

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»**

**Факультет інформатики та обчислювальної техніки  
Кафедра автоматики та управління в технічних системах**

«На правах рукопису»  
УДК \_\_\_\_\_004.622\_\_\_\_\_

До захисту допущено:  
Завідувач кафедри  
\_\_\_\_\_ Олександр РОЛІК  
«\_\_» \_\_\_\_\_ 20\_\_ р.

**Магістерська дисертація  
на здобуття ступеня магістра  
за освітньо-професійною програмою «Програмне забезпечення інформаційно-  
комунікаційних систем»  
зі спеціальності 121 «Інженерія програмного забезпечення»  
на тему: «Система онлайн-страхування та обліку сервісних робіт  
автотранспорту»**

Виконав (-ла):  
студент (-ка) VI курсу, групи ІТ-393мп  
Петрук Євген Сергійович \_\_\_\_\_

Керівник:  
завідувач кафедри АУТС, д.т.н., професор  
Ролік Олександр Іванович \_\_\_\_\_

Рецензент:  
декан факультету прикладної математики, д.т.н., професор  
Дичка Іван Андрійович \_\_\_\_\_

Засвідчую, що у цій магістерській дисертації  
немає запозичень з праць інших авторів без  
відповідних посилань.  
Студент \_\_\_\_\_

**Національний технічний університет України**  
**«Київський політехнічний інститут імені Ігоря Сікорського»**  
**Факультет інформатики та обчислювальної техніки**  
**Кафедра автоматики та управління в технічних системах**

Рівень вищої освіти – другий (магістерський)

Спеціальність – 121 «Інженерія програмного забезпечення»

Освітньо-професійна програма «Програмне забезпечення інформаційно-комунікаційних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

\_\_\_\_\_ Олександр РОЛІК

«\_\_» \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**  
**на магістерську дисертацію студенту**  
**Петруку Євгене Сергійовичу**

1. Тема дисертації «Система онлайн-страхування та обліку сервісних робіт автотранспорту», науковий керівник дисертації Ролік Олександр Іванович, завідувач кафедри АУТС, д.т.н., професор, затверджені наказом по університету від «26» 10 2020 р. №3132-с

2. Термін подання студентом дисертації \_\_\_\_\_

3. Об'єкт дослідження: система онлайн-страхування та обліку сервісних робіт автотранспорту побудована з використанням архітектурі хмарного сервісу AWS

4. Вихідні дані: система онлайн-страхування та обліку сервісних робіт автотранспорту

5. Перелік завдань, які потрібно розробити: огляд існуючих рішень та систем для реалізації страхових продуктів онлайн, огляд ймовірних платформ для розробки системи, налаштування середовища розробки та проектування системи, проектування бази даних системи, проектування оптимальної інфраструктури з використанням хмарного сервісу AWS, проектування та розробка клієнтського інтерфейсу системи, розробка серверної частини.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу

7. Орієнтовний перелік публікацій: Акт впровадження ТОВ «Аудиторська компанія «Крюнер і ко»»

8. Дата видачі завдання 29 квітня 2020

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Збір та обробка інформації	01.06.2020	
2.	Розробка змісту магістерської дисертації та узгодження з науковим керівником	12.08.2020	
3.	Огляд існуючих рішень та підготовка звіту	16.09.2020	
4.	Проектування системи	03.10.2020	
5.	Програмна реалізація системи	20.10.2020	
6.	Тестування та аналіз роботи системи	15.11.2020	
7.	Розробка стартап-проекту	20.11.2020	
8.	Оформлення магістерської дисертації	25.11.2020	
9.	Попередній захист	02.12.2020	
10.	Захист	22.12.2020	

Студент

Євген ПЕТРУК

Науковий керівник

Олександр РОЛІК

## РЕФЕРАТ

Магістерська дисертація: 100с., 40 рис., 25 табл., 8 додатків, 22 джерела.

Актуальність проблеми. Будь-хто з нас, кому доводилося страхувати своє майно або здоров'я у агентів або в офісах компаній, знає, що ця процедура забирає багато часу і сил. Особливо страждають власники автотранспорту, яким щорічно доводиться продовжувати ОСАГО. Незважаючи на видиму простоту цього процесу і чіткі стандарти в області тарифів, на оформлення поліса може піти кілька годин.

Багато страхових компаній в останні роки пропонують спростити страхування і оформити поліс онлайн. Високі технології не стоять на місці – зараз більшості жителів України доступні можливості Інтернету. Вийти в мережу можна не тільки з комп'ютера, але і з інших гаджетів - наприклад, мобільного телефону. І не дивно, що слідом за іншими системами страхові компанії також пішли назустріч своїм клієнтам, надавши можливість користуватися послугами в режимі онлайн.

Якщо зовсім недавно за допомогою Інтернету ми могли тільки зайти в особистий кабінет і переглянути інформацію про страхові внески та оформлених полісах, то в останні роки страхові компанії зробили крок далі - дозволили клієнтам страхуватися в онлайн-режимі.

Разом з тим онлайн-страхування дає реальну можливість скоротити витрати часу і сил, які раніше були необхідні для візиту в страхову компанію і очікування своєї черги. Багато видів полісів взагалі не припускають необхідність спілкування з фахівцями – зробити їх оформлення можна повністю в віддаленому режимі. Вище викладений матеріал підтверджує актуальність проблеми та перспективність розробки системи онлайн-страхування та обліку сервісних робіт автотранспорту.

Мета і задачі дослідження. Метою роботи є автоматизація та вдосконалення процесу формування і реалізації страхових полісів, що дасть змогу розраховувати конкурентну вартість страхових продуктів шляхом врахування всіх можливих коефіцієнтів та зменшення штату працівників.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- дослідити сферу онлайн-страхування;

- визначити основних гравців на ринку;
- дослідити переваги та недоліки існуючих рішень;
- розглянути основні технології для створення веб-додатків;
- спроектувати базу даних для обслуговування системи;
- спроектувати та розробити Frontend частину системи;
- спроектувати та розробити Backend частину системи;
- розробити стартап-проект відповідно до теми дисертації.

Об'єкт дослідження. Система онлайн-страхування та обліку сервісних робіт автотранспорту побудована з використанням архітектури хмарного сервісу AWS.

Предмет дослідження. Методи та підходи реалізації конкурентної системи онлайн-страхування та продажу сервісних робіт автотранспорту.

Практична цінність. Отримані засоби реалізації системи онлайн-страхування та обліку автотранспорту можуть бути використані у майбутніх дослідженнях за напрямками:

- розвиток онлайн-страхування та технологій автоматичних продажів у веб-системах;
- вдосконалення системи на базі фреймворку Laravel.

Ключові слова. Онлайн-страхування, калькулятор страховки, веб-система, ReactJS, JavaScript, MySQL, Laravel, Frontend, Backend

## ABSTRACT

Master's dissertation: 100 p., 40 figures, 25 tables, 8 supplement, 22 sources.

The urgency of the problem. Any of us who have had to insure our property or health from agents or company offices knows that this procedure takes a lot of time and effort. Owners of motor vehicles, who have to continue CTP every year, suffer especially. Despite the apparent simplicity of this process and clear standards in the field of tariffs, the policy can take several hours.

Many insurance companies in recent years offer to simplify insurance and issue a policy online. High technologies do not stand still - now most Ukrainians have access to the Internet. You can go online not only from a computer, but also from other gadgets - such as a mobile phone. Not surprisingly, following other systems, insurance companies have also met the needs of their customers by providing the opportunity to use the services online.

If recently with the help of the Internet you could only go to the personal account and view information about insurance premiums and issued policies, in recent years, insurance companies have taken a step further – allowed customers to insure online.

However, the new service has not found the proper popularity – many customers are simply afraid to draw up important documents remotely, preferring personal communication with the insurance agent

However, online insurance provides a real opportunity to reduce the time and effort that were previously required to visit the insurance company and wait your turn. Many types of policies do not involve the need to communicate with professionals – you can make their registration completely remotely. The above material confirms the urgency of the problem and the prospects for the development of an online insurance system and accounting for service work of vehicles.

Purpose and tasks of the research. The purpose of the work is to automate and improve the process of formation and implementation of insurance policies, which will allow to calculate the competitive value of insurance products by considering all possible coefficients and reducing the number of employees.

To achieve this goal, it is necessary to solve the following tasks:

- explore the field of online insurance;
- identify the main players in the market;
- explore the advantages and disadvantages of existing solutions;
- consider the basic technologies for creating web applications;
- design a database for system maintenance;
- design and develop the Frontend part of the system;
- design and develop a Backend part of the system;
- develop a startup project in accordance with the topic of the dissertation.

Object of study. Online insurance and vehicle maintenance registration system is constructed with use of architecture of cloud service AWS.

Subject of study. Methods and approaches of realization of competitive online insurance and vehicle maintenance registration system

Practical value. The obtained means of implementation of the online insurance and vehicle maintenance registration system can be used in future research in the areas:

- development of online insurance and automatic sales technologies for web systems;
- system improvement based on the Laravel framework.

Keywords. Online insurance, insurance calculator, web system, ReactJS, JavaScript, Laravel, Frontend, Backend.

## Зміст

ВСТУП.....	10
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ .....	12
1.1 Поняття інтернет-страхування .....	12
1.2 Використання інтернет-каналів страховими компаній.....	13
1.3 Види страхування онлайн .....	15
1.4 Принципи роботи онлайн страхування .....	16
1.5 Переваги та недоліки страхування через мережу інтернет .....	17
1.6 Сучасна ситуація на ринку.....	18
1.7 Постановка задачі .....	22
2 ОГЛЯД СУЧАСНИХ РІШЕНЬ .....	24
2.1 Oh.UA.....	24
2.2 Finscanner.com .....	25
2.3 Polismart.com.ua .....	27
2.4 Висновки.....	29
3 РОЗРОБЛЕННЯ СИСТЕМИ ОНЛАЙН-СТРАХУВАННЯ ТА ОБЛІКУ СЕРВІСНИХ РОБІТ АВТОТРАНСПОРТУ .....	32
3.1 Формули для обчислення вартості страхових полісів системи .....	32
3.2 Розробка системи.....	33
3.2.1 Модуль підбору страхової компанії та купівлі полісу .....	34
3.2.2 Модуль обліку сервісних робіт та витрат автотранспорту .....	38
3.3 Встановлення програмного забезпечення та налаштування середовища розробки .....	48
3.3.1 Встановлення та налаштування платформи віртуалізації Oracle VM VirtualBox.....	48
3.3.2 Встановлення Ubuntu версії 20.04 .....	49
3.3.3 Встановлення локального web-сервера Apache2 .....	58
3.3.4 Встановлення Framework Laravel .....	61



3.4	Проектування та налаштування інфраструктури AWS .....	63
3.5	Проектування бази даних.....	67
3.6	Розробка Frontend .....	71
3.7	Розробка Backend.....	76
4	РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ.....	83
4.1	Опис ідеї проекту.....	83
4.2	Технічний аудит ідеї проекту .....	84
4.3	Аналіз ринкових можливостей запуску стартап-проекту .....	85
4.4	Розроблення маркетингової програми стартап-проекту.....	93
4.5	Висновки.....	96
	ВИСНОВКИ.....	97
	ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	99

## ВСТУП

Сучасний рівень розвитку системи обробки інформації завершує етап еволюції ІТ-індустрії [1], що повертаються до зосереджених центральних обробок інформації.

Інформація, яка в 60-х – 80-х роках минулого століття існувала у вигляді обчислювальних центрів колективного користування (ОЦКК), а в наш час – в формі центрів обробки даних (ЦОД), що представляють собою сукупність інформаційної, телекомунікаційної та інженерної інфраструктури. При цьому ми стали свідками чергового витка еволюції ІТ, показником якого є повсюдне використання хмарних обчислень, реалізованих на основі ЦОД. Прийняття парадигми хмарних обчислень визначає формування нового ІТ-середовища, яке складає основу формуючого ІТ-суспільства. І як результат відкриває нові можливості вже для існуючих сфер бізнесу, наприклад, таких як страхування.

Online-покупки вже давно увійшли в життя сучасної людини: сьогодні в режимі онлайн можна купити все що завгодно— від квитків до продуктів харчування.

Щороку все більше і більше фінансових установ використовують Інтернет для надання своїх послуг. Першими були банки та інші інвестиційні посередники, а тепер до них приєдналися страхові компанії.

Незважаючи на те, що український ринок онлайн-страхування ще досить молодий, він уже має досить велику кількість страховиків, які так чи інакше надають свої послуги через Інтернет. Природно, як і для будь-якої іншої форми електронного бізнесу, ринок Інтернет-страхування [3] найбільш розвинений у Європі та США.

Наприклад, у Великобританії близько 60% страхування продається через Інтернет, тоді як в Україні - менше 1%. Однак українські страховики сповнені оптимізму та пророкують велике майбутнє такого типу торгівлі.

За оцінками експертів, за останні кілька років обсяг страхування, проданого українськими компаніями через Інтернет, зріс більш ніж удвічі. У 2019 році продажі за цим каналом продовжують зростати ще активніше, ніж у минулому.

Дійсно, як зазначалося раніше, на всьому страховому ринку продажі в Інтернеті все ще займають низьку частку - менше 1%. І навіть більшість з них - це не інтернет-магазини, де ви можете вибрати, оплатити та придбати поліс, а окремі послуги на веб-сайтах страховиків, за допомогою яких людина може лише вибрати страховку, розрахувати її вартість та замовлення.

Страховики стверджують [2], що сьогодні через Інтернет можна придбати майже будь-який вид страхування, що пропонується фізичним особам - різні види страхування КАСКО, квартири, будинки, страхування ремонту, поліси ОСВ, добровільні поліси страхування відповідальності автотранспортних засобів, страхування від нещасних випадків і навіть медичне обслуговування страхування.

Враховуючи все вищесказане, важливість вивчення ринку Інтернет-страхування дуже велика, враховуючи зростаючий попит на цей вид послуг, система, яка розроблена пізніше в цій дисертації, має унікальний набір функціональних можливостей для Розрахунків низької вартості поліса і є конкурентною платформою для вибору та придбання необхідного виду страхового полісу для автомобіля в Інтернеті та обліку робіт з технічного обслуговування автомобіля.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1 Поняття інтернет-страхування

Перш ніж говорити про інтернет-страхування, згадаємо в загальних рисах про те, що з себе представляє традиційне страхування.

Страхування [4] – це домовленість зі страховою компанією, згідно з якою клієнт платить їй регулярні суми коштів, і вона погоджується покрити його витрати, якщо трапиться певна нещасна подія, наприклад дорожньо-транспортна пригода, пошкодження майна або хвороба.

Однак страховка також може бути організована на загальний підставах, наприклад якщо клієнт досягнете певного віку. Подія, що охоплюється страхуванням, називається страховою подією. Особа, у якої така подія відбувається або може відбутися, називається застрахованою. Коли настає страховий випадок (ви захворіли, потерпіли нещасний випадок, досягли певного віку), страховик виплатить вам певну суму грошей, яку називають страховою виплатою / виплатою. Виплачена страхова допомога / вимога допоможе вам або вашій родині подолати фінансові труднощі або збільшені витрати, які можуть виникнути внаслідок страхового випадку. Це означає, що, організувавши страхування, ви отримуєте страхове покриття від певного ризику.

Під час операції формується документ, який називається страховим полісом. Поліс є юридичним документом для Страхувальника та для страхової компанії, визначає основні моменти страхування: об'єкт страхування (майно, особа, відповідальність тощо), страховий випадок, з моменту укладення договору, початок і кінець страхового періоду, сума страхування, страховий збір. Документ підписаний обома сторонами і повинен зберігатися у Страхувальника.

Таким чином, Інтернет-страхування (у повному розумінні цього слова) - це всі перераховані вище елементи взаємодії між страховою компанією та клієнтом, що виникають внаслідок продажу страхового продукту та його обслуговування, але здійснюються за допомогою Інтернету. Отже, щоб онлайн-представництво компанії могло функціонувати як віртуальний офіс цієї страхової компанії, воно повинно виконувати всі функції звичайного офісу.

Незалежно від типу онлайн-страховки, на які потребує клієнт, дуже важливо спочатку отримати про це знання. Наприклад, якщо потрібне автострахування, клієнт повинен знати різницю між сторонніми та всеохоплюючими онлайн-страховими планами.

Аналогічно, для строкового страхування замовники повинні знати, як це працює, та фактори, що впливають на премію полісу.

Змінивши парадигму продажів на Інтернет, індустрія страхування в Україні змінила гру для страхових клієнтів. Зараз страховики пропонують -страхові продукти онлайн, починаючи від строкового страхування, накопичувальних планів та пенсійних планів.

Для кінцевого клієнта, страхові поліси онлайн доступні за допомогою декількох клацань миші або натискань на їх смартфонах. Крім того, весь процес придбання онлайн-страховки відрізняється від придбання таких же програм через місцевого агента. Замовник можете придбати страховий поліс з будь-якого місця і в будь-який час.

Найголовніше те, що є можливість придбання страхового плану онлайн полегшує клієнту прийняття обґрунтованого рішення. Легше порівняти різні плани страхування в Інтернеті з точки зору премії та пропонованих вигод, отриманих від процесу реалізації страхування в Інтернеті. Як тільки клієнт зважить усі «за» і «проти», він може вибрати правильний план, який відповідає його потребам.

Поряд із придбанням страховки в Інтернеті, клієнт можете скористатися різними супутніми послугами, такими як завантаження документа про поліс, отримання онлайн-страхових котирувань та оновлення полісу в Інтернеті.

## 1.2 Використання інтернет-каналів страховими компаніями

Більшість із страхових компаній пропонують виключно інформацію про себе та послуги, які вони пропонують. Лише незначна частина страхових компаній має на своїх веб-сайтах програми для підтримки та збільшення продаж як калькулятори, квоти тощо.

Більша частина інформації, що представлена в Інтернеті це звичайні сайти страхових компаній які мають рекламну роль і спрямовані на впровадження страхового бізнесу та послуг, які пропонують компанії- страховики.

Рівень інтерактивності, який пропонує веб-сайт, може залучити або відвернути відвідувачів. Можна досягти інтерактивності завдяки впровадженню посилань електронною поштою, що дає можливість онлайн-зв'язку зі спеціалістами компанії або через архівування досвіду фактичних споживачів продукції та послуг компанії. Видно, що рівень реалізованої інтерактивності між споживачем та страховою компанією не на бажаному рівні.

Можна помітити, що є страхові компанії, які не пропонують контакт з електронною поштою як одне з основних засоби Інтернет-спілкування. Однак, щоб не ігнорувати існування раніше згаданих практик, в середньому, більшість українських страхових компаній пропонують клієнтам можливість з ними зв'язатися електронною поштою, і вони мають вікна на своїх сайтах для конкретних питань, адресованих відповідним секторам (відділи) у страховій компанії.

Українські страхові компанії, як правило, покладаються на прямий продаж шляхом фізичного розподілу страхових полісів.

Деякі зі страхових компаній пропонують можливість електронних планування візиту страхових агентів. Звичайно, це лише у функції підтримки та збільшення продаж за допомогою фізичного розповсюдження та не стосується поширення продуктів через Інтернет. Новаторські кроки до електронної комерції все ще є короткими і служать для збільшення, підтримки та просування традиційний фізичний розподіл страхових продуктів, що є набагато дорожчим видом розподілу, як він раніше згадувалося в цій роботі. Розширений рівень електронної комерції - це впровадження електронної транзакції, яка включає купівлю та продаж продуктів та послуги з використанням Інтернету, що підтримуються грошовим переказом. Цей рівень взаємодії зі споживачами здійснює дуже незначна кількість компаній в Україні.

Сучасний рівень можливостей транзакцій є справжньою електронною комерцією, і потенційні вигоди як для споживачів, так і для постачальників величезні, якщо їх належним чином впровадити.

Страховання онлайн не має повністю автоматизованої електронної комерції на даний момент, оскільки воно включає в самому широкому розумінні отримання електронного сертифіката придбаного поліса.

Поліс або товар, що купується, досі доставлений фізично. Виходячи з даних, що отримані від замовників страхового полісу, засвідчують, що такий спосіб викупу та оплати, хоча і впроваджується лише останнім, свідчить про високий ступінь привабливості.

Більше того, це займає близько 30% від загального продажу полісів за 2019 рік і, як вже було зазначено раніше, це онлайн-канал доступний лише для незначної кількості страхових агентів. Цей показник є очікуваним тому що це слово для дуже частих страхових послуг, які адаптуються до електронного бізнесу.

### 1.3 Види страхування онлайн

На даний момент в Україні практикуються наступні види інтернет-страхування:

- ОСАЦВ (страхування цивільної відповідальності автомобілістів);
- КАСКО;
- майнове страхування;
- страхування життя;
- медичні страховки;
- страхування від нещасного випадку.

ОСАЦВ – це страхування цивільної відповідальності автовласників перед іншими особами. Воно застосовується в разі, коли використання автомобіля заподіяло шкоду іншим громадянам (їх здоров'ю, майну або навіть життю). Відповідно до українського законодавства, ОСАЦВ є обов'язковим.

КАСКО – вид страхування, при якому автомобіль страхується від різних можливих випадків псування або втрати (викрадення).

Страхування майна онлайн в основному практикується для страхування квартир, будинків та громадянської відповідальності перед сусідами.

Страхування життя – страхування, яке передбачає захист інтересів застрахованої особи, пов'язаних з його життям і смертю. Страхування життя, як правило, пов'язане з довгостроковими інтересами страхувальника/застрахованої особи в силу того, що життя розглядається як тривалий стан, і, відповідно, подія смерті бачиться непрогнозованою та віддаленою.

Медичне страхування - це вид страхового покриття, який зазвичай оплачує медичні, хірургічні, лікарські засоби, а іноді і стоматологічні витрати, понесені страхувальником. Медичне страхування може відшкодувати страхувальнику витрати, понесені внаслідок хвороби або травми, або ж безпосередньо оплатити доглядача.

Онлайн страхування від нещасних випадків пропонується зараз як для фізичних, так і для юридичних осіб. Приватні особи можуть придбати електронну страховку на випадки втрати тимчасової непрацездатності, для оплати лікарських препаратів. Для юридичних осіб страхові компанії пропонують страхування від нещасних випадків для всіх співробітників їхніх фірм. Такий поліс допомагає відшкодувати збитки при настанні нещасних випадків на робочому місці.

#### 1.4 Принципи роботи онлайн страхування

Зазвичай є 5 кроків, щоб подати заявку на отримання поліса онлайн:

- вибрати страховий продукт однієї з страхових компаній;
- обрати кращу пропозицію для розрахунку страхової вартості, та суми відшкодування і супутніх послуг;
- подати заявку, вказавши всі обов'язкові поля запропоновані форми;
- сплатити страховий внесок страховика готівкою або перерахувати його на рахунок електронним платежем, кредитною картою або інше;
- отримати поліс від агента або кур'єром чи поштою (звичайною або електронною).



Купівля вашої страховки в Інтернеті пропускає посередника, який є страховим агентом. Пряма покупка страховки повинна заощадити гроші, оскільки страховій компанії не потрібно платити агенту.

### 1.5 Переваги та недоліки страхування через мережу інтернет

Як і будь-який інший вид надання послуг, інтернет-страхування має переваги і недоліки. Головними позитивними рисами для компанії є:

- обслуговування віртуального офісу набагато дешевше, тому що компанія економить на оренді приміщення, комісійних винагородах тощо, знижуються транзакційні витрати, які потрібні для обслуговування клієнта в звичайному офісі;
- можливість збільшення кількості потенційних клієнтів;
- Рамки просування страхових послуг через інтернет визначаються не державними кордонами, а поширенням інтернету в світі;
- можливість цілодобового обслуговування клієнтів;
- збільшення конкурентоспроможності;
- пропонуючи свої послуги через всесвітню мережу, компанії роблять їх більш доступними, зрозумілими і зручними для клієнтів;
- поліпшення іміджу;
- наявність у компанії зручного, функціонального, красивого віртуального офісу говорить про серйозність і професіоналізм даної організації, тим самим переконуючи себе довіру клієнтів.

Позитивні аспекти для страхувальника:

- економія часу – для ознайомлення з послугами і цінами страхових компаній не доведеться їхати в офіс і витрачати свій особистий час, досить зайти на сайт компанії і подивитися потрібну інформацію, а також так само можна без тимчасових витрат отримати страховий поліс;
- можливість оптимального вибору певних видів послуг у різних страхових компаній – на сайті клієнт може отримати повну інформацію зі свого питання і порівняти її з різними пропозиціями страховиків;

– економія коштів – на більшість послуг інтернет-страхування надаються знижки (це викликано тим, що страховики економлять власні кошти на інтернет-торгівлі).

Але поряд із позитивними моментами виникають і певні складнощі:

– не всі види страхування можна надати через інтернет. Наприклад, майнове страхування. Багато об'єктів вимагають огляду страховика, такі як великі підприємства, елітні будинки, щоб точно впевнитися в дійсній вартості майна;

– багато громадян, а саме в Україні, проявляють недовіру до страхових компаній. На це вплинуло безліч причин. Але погіршує ситуацію і велика ймовірність страхового інтернет-шахрайства. Так само певну роль грає невпевненість в тому, що без особистої участі клієнта страхова компанія зможе гарантувати правильність укладення договору, стабільну доставку поліса і збереження прав клієнта;

– не у всіх країнах діє електронно-цифровий підпис і тому страховику доведеться доставляти страховий поліс і договір страхування через кур'єра або поштою, що спричинить додаткові витрати, особливо у випадку великої віддаленості страхувальника від страховика;

– не всі споживачі мають комп'ютер або можливість виходу в глобальну мережу, а частина з них не вміє нею користуватися;

– для того щоб почати інтернет-страхування потрібні великі капіталовкладення, тому ця сфера доступна тільки великим компаніям.

## 1.6 Сучасна ситуація на ринку

За 2019 рік світовими лідерами у страхуванні зібрано страхових премій на значні суми. У Росії та Україні ринок страхування знаходиться в процесі розвитку, що наочно демонструє табл. 1.1

Таблиця 1.1 – Обсяг страхових премій із сегментацією за країнами

Країна	Обсяг зібраних страхових премій, млн дол. США	Питома вага у світі, %
США	1204677	26,77
Японія	655408	14,26

Великобританія	319553	6,96
Франція	273112	5,94
Німеччина	245162	5,34
Росія	43257	0,94
Україна	3084	0,07

Таких обсягів високо розвинутим країнам вдається досягти, у т.ч. за рахунок впровадження страхового маркетингу у мережі Інтернет, який давно практикується і приносить до 60% страхових внесків.

За допомогою власного корпоративного сайту страхова компанія може покращити свій імідж, реалізувати страхові поліси, надавати повний спектр інформаційних та консультаційних послуг, підтримувати постійний контакт не тільки зі своїми клієнтами, а й з діловими партнерами, надавати додаткові послуги і проводити необхідні дослідження з використанням інтернет-ресурсів, наприклад, для пошуку потенційних клієнтів та формування санкціонованого списку електронних адрес майбутніх страхувальників

В Україні, згідно з дослідженнями InMind та міжнародної мережі Factum-Group, кількість населених пунктів, що мають доступ до Інтернету, серед населених пунктів України віком від 15 років становить 13,1 млн (на початок 2018 року).

На початок 2019 року кількість користувачів Інтернету у віці 15+ досягла 18,3 млн., А вікова структура така: від 15 до 29 років - 44%, 30-39 - 35%, старше 45 років - 24%. . За суб'єктивним доходом користувачі Інтернету розподіляються наступним чином: малозабезпечені - 3%; нижче середнього - 22%; в середньому - 57%; вище середнього - 18%.

Таким чином, протягом останньої години ми спостерігаємо тенденцію до збільшення аудиторії користувачів Інтернету серед населених пунктів України, що є потенційною клієнтською базою для страховиків.

В минулому році компанії (AXA Insurance, ASKA, ORANTA, INGO Ukraine, Alpha Insurance, TAS, Universal) мають власні торгові таблиці. 1.2 покращений Інтернет, новітні недорогі канали розподілу своєї продукції..



	Обов'язкове страхування цивільно-правової відповідальності		+	+	+									
	КАСКО	+	+	+	+		+	+	+	+	+	+		
	Страхування «Зелена картка»		+	+					+	+	+	+	+	
	Медичне добровільне страхування		+	+	+	+				+				
	Страхування від нещасних випадків	+	+	+			+							
	Страхування майна	+	+	+	+		+			+				
	Страхування фінансів			+										
	Страхування відповідальності власників собак		+											
	Туристичне страхування		+		+	+	+		+	+		+		+

Більшість із зазначених вище страховиків почали продавати страхові продукти в Інтернеті переважно в 2013 році. Однак, все ще існує багато компаній з найбільшими сумами валових страхових внесків, які не надають онлайн-страхових послуг.

Згідно з дослідженням, проведеним Swiss Re, адміністративні витрати на укладення страхового договору за допомогою прямих продажів за прямими каналами продажу складають до 19 доларів за транзакцію, при використанні телефону - до 8 доларів за транзакцію, в Інтернеті - до 0,5 доларів . за транзакцію (або інші розрахунки, вартість страхового продукту при покупці через Інтернет знижується в

середньому на 20-25%). Можна припустити, що інноваційний метод продажу страхових полісів в Україні також отримуватиме до 65% річних, як це було за кордоном вже багато років, але для визначення чинності ЕЦП потрібне регуляторне законодавство.

Також факторами, що заважають розвитку Інтернет-страхування в Україні, є:

- низький рівень доходів страхувальників; невелика частка користувачів Інтернету в порівнянні з провідними країнами світу;
- слабкий рівень фінансової грамотності населення та забезпечення електронних платіжних систем в Україні;
- недовіра до сайтів та низький рівень страхової культури населення.

Враховуючи наведені данні можна перейти до постановки задачі де мають бути враховані ці чинники.

### 1.7 Постановка задачі

Пересічному громадянину важко зорієнтуватися у великій кількості пропозицій та полісів, що пропонують страхові компанії на сьогоднішній день. Тому цілком зрозуміло, що постає питання необхідності розробки програмного забезпечення, яке б допомогло потенційному клієнту обрати саме те, що йому потрібно не докладаючи для цього зайвих зусиль.

Особливо гостро стоїть питання вибору страхової компанії та полісу для власників автотранспорту, оскільки цей вид страхування є обов'язковим в Україні. Крім того, деякі транспортні засоби (наприклад, автомобілі, що використовуються у якості таксі) підлягають обов'язковому технічному контролю, офіційної системи про нагадування проходження якого на сьогоднішній день в Україні немає.

Тому метою магістерської дисертації є автоматизація та вдосконалення процесу формування і реалізації страхових полісів, що дасть змогу розраховувати конкурентну вартість страхових продуктів шляхом врахування всіх можливих коефіцієнтів та зменшення штату працівників.

Таким чином, система буде складатися з двох модулів: модуля підбору страхової компанії і купівлі страхового полісу та модуля обліку сервісних робіт і витрат транспортного засобу, що покликані вирішити наступні питання клієнта:

1. Модуль підбору страхової компанії та купівлі полісу:

- вибір надійної страхової компанії за рейтингом;
- порівняння страхових компаній;
- купівля страхового полісу онлайн за найбільш вигідною;
- вибір способу доставки полісу у будь-яку точку України.

2. Модуль обліку сервісних робіт та витрат автотранспорту:

- налаштування графіку обов'язкового технічного контролю;
- нагадування про необхідність проведення сервісних робіт;
- слідкування за станом авто;
- оновлення пробігу для виконання регламентних робіт на його основі;
- облік витрат на транспортний засіб (витрати на паливо, ремонт тощо);
- список точок-партнерів, де можна здійснювати на пільгових умовах ремонт, шиномонтаж та інші види сервісних робіт, з адресами, маршрутами і можливістю бронювання;
- максимально швидке та зручне бронювання, замовлення та онлайн оплата послуг у вищезазначених точках.

## 2 ОГЛЯД СУЧАСНИХ РІШЕНЬ

### 2.1 Oh.UA

Oh.UA – online-сервіс для порівняння і швидкого пошуку, а також придбання страхових полісів від страхових компаній України, що є партнерами проекту.

На сайті є онлайн-калькулятор для розрахування наступних видів страховок:

- ОСАЦВ;
- КАСКО;
- зелена карта;
- туристичне страхування.

На рис. 2.1 можна побачити, як виглядає інтерфейс [5] онлайн-калькулятора на сайті Oh.UA для розрахування вартості полісу обов'язкового страхування автоцивільної відповідальності (ОСАЦВ).

Тип транспортного средства

Город регистрации

☐ Используется как такси

НАЙТИ ПРЕДЛОЖЕНИЯ

[ОСАГО в Киеве](#) | 
 [ОСАГО в Одессе](#) | 
 [ОСАГО в Харькове](#) | 
 [ОСАГО во Львове](#)

[Калькулятор ОСАГО](#) | 
 [Автогражданка](#)

Легковой до 1600 см3  
 Легковой до 1600 см3  
 Легковой - 1601 - 2000 см3  
 Легковой - 2001 - 3000 см3  
 Легковой - более 3000 см3  
 Грузовой - до 2т  
 Грузовой - более 2т  
 Автобус - до 20 человек  
 Автобус - более 20 человек  
 Прицеп к грузовому авто  
 Прицеп к легковому авто  
 Мотоцикл / мотороллер до 300 см3  
 Мотоцикл / мотороллер более 300 см3

Рисунок 2.1 – Інтерфейс онлайн-калькулятора проекту Oh.UA для розрахування вартості полісу ОСАЦВ.

Для того, щоб скористатися калькулятором необхідно вказати:

- тип вашого автомобіля та об'єм двигуна;
- місце реєстрації;
- чи використовується транспортний засіб в якості таксі.



Після заповнення користувачем необхідних полів онлайн-калькулятор проекту Oh.UA самостійно розрахує за вказаними параметрами вартість полісу ОСАЦВ в різних страхових компаніях-партнерах (рис. 2.2).

КОМПАНІЯ	РЕЙТИНГ	ФРАНШИЗА	ЦЕНА	БОНУСИ	
<b>UNIVES</b> О компанія	ОСАГО от СК «Фонбет» Рейтинг страховщика: 80%	1000 грн	<del>1044</del> грн 707 грн	минус 32% от Oh.UA Технический ассистент	КУПИТЬ
<b>VUSO</b> О компания	ОСАГО от СК «ВУСО» Рейтинг страховщика: 96%	0 грн	<del>1037</del> грн 882 грн	минус 15% от Oh.UA Выбор Клиентов полное урегулирование Бесплатная доставка	КУПИТЬ
<b>ПРОВІДНА</b> О компания	ОСАГО от СК «Провідна» Рейтинг страховщика: 94%	0 грн	<del>1037</del> грн 892 грн	минус 15% от Oh.UA Технический ассистент Выбор Клиентов полное урегулирование	КУПИТЬ
<b>АС   Альфа Страхування</b> О компания	ОСАГО от СК «Альфа Страхування» Рейтинг страховщика: 90%	0 грн	<del>1037</del> грн 903 грн	минус 13% от Oh.UA Бесплатная доставка	КУПИТЬ

Рисунок 2.2 – Результат розрахування вартості полісу ОСАЦВ в онлайн-калькуляторі на сайті Oh.UA

## 2.2 Finscanner.com

Finscanner – сервіс, що надає клієнтові інформацію про фінансові послуги компаній та надає можливість провести їх порівняння, щоб обрати ту послугу, яка найбільш підходить за умовами і вартості.

В даний момент сервіс працює лише зі страховими компаніями. Як обіцяють автори проекту, згодом Finscanner планує запустити ще кілька фінансових послуг, які дозволять вирішувати й інші щоденні фінансові питання.

На сайті є онлайн-калькулятор для розрахування наступних видів страховок:

- ОСАЦВ;
- зелена карта;
- туристичне страхування.

На рис. 2.3 можна побачити, як виглядає інтерфейс онлайн-калькулятора на сайті Finscanner.com.

The screenshot shows the 'Шаг 1 Ввод данных' (Step 1: Enter data) section of the Finscanner website. It features a navigation bar with three steps: 'Шаг 1 Ввод данных' (highlighted in blue), 'Шаг 2 Выбор страхового полиса' (Selection of insurance policy), and 'Шаг 3 Покупка' (Purchase). Below the navigation bar, there are five vehicle type options with icons: 'Легковой автомобиль' (highlighted with a blue border), 'Электромобиль', 'Грузовой автомобиль', 'Пассажирский автобус', and 'Мотоцикл, мотороллер'. Under these, there are four engine volume options: 'Объем двигателя: До 1,6 л включительно' (selected with a blue dot), '1,601 - 2,0 л включительно', '2,001 - 3,0 л включительно', and '3,001 л и более'. Below the engine options, there is a 'Город регистрации:' (Registration city) dropdown menu with 'Киев' selected, and other options: 'Одесса', 'Харьков', 'Днепр', 'Запорожье', and 'Другой город'. At the bottom, there are two more dropdowns: 'Водительский стаж: Менее 3 лет' (selected) and 'Безаварийный стаж вождения: До 1' (selected). A large orange button labeled 'Найти предложения' (Find offers) is located at the bottom right.

Рисунок 2.3 – Интерфейс онлайн-калькулятора проекту Finscanner для розрахування вартості полісу ОСАЦВ

Для того, щоб скористатися калькулятором необхідно вказати:

- тип вашого транспортного засобу;
- об'єм двигуна;
- місце реєстрації;
- стаж водіння (в т.ч. безаварійний).

Результат пошуку зображений на рис. 2.4.

Сортировать по: Рейтингу   Цена   ☐ С франшизой (12) ☐ Только акционные (11)




<p><b>Скидка 10%</b></p>  <p><b>Княжа</b></p> <p>FinScanner рейтинг: ★★★★★</p> <p><a href="#">Подробный рейтинг</a></p>	<p>Франшиза:</p> <p><input checked="" type="radio"/> 0 грн <input type="radio"/> 1000 грн</p> <p><del>1 296 грн</del> <b>1 167 грн</b></p> <p><b>Оформить</b></p>	<p>Бонусы:</p> <ul style="list-style-type: none"> <li>+ Консьерж-поддержка FinScanner 24/7</li> <li>+ Бесплатная доставка по Киеву</li> </ul> <p><a href="#">Смотреть все условия</a></p>
<p><b>Скидка 5%</b></p>  <p><b>PZU Украина</b></p> <p>FinScanner рейтинг: ★★★★★</p> <p><a href="#">Подробный рейтинг</a></p>	<p>Франшиза:</p> <p><input checked="" type="radio"/> 0 грн <input type="radio"/> 1000 грн</p> <p><del>1 296 грн</del> <b>1 232 грн</b></p> <p><b>Оформить</b></p>	<p>Бонусы:</p> <ul style="list-style-type: none"> <li>+ Консьерж-поддержка FinScanner 24/7</li> <li>+ Бесплатная доставка по Киеву</li> <li>+ PZU Assistance 24/7</li> </ul> <p><a href="#">Смотреть все условия</a></p>
<p><b>Скидка 25%</b></p>  <p><b>Европейский Страховой Альянс</b></p> <p><a href="#">Подробный рейтинг</a></p>	<p>Франшиза:</p> <p><input checked="" type="radio"/> 510 грн <input type="radio"/> 1000 грн</p> <p><del>1 296 грн</del> <b>1 167 грн</b></p> <p><b>Оформить</b></p>	<p>Бонусы:</p> <ul style="list-style-type: none"> <li>+ Консьерж-поддержка 24/7</li> </ul> <p><a href="#">Смотреть все условия</a></p>

Рисунок 2.4 – Результат розрахування вартості полісу ОСАЦВ в онлайн-калькуляторі на сайті Finscanner.com

### 2.3 Polismart.com.ua

Polismart – це сервіс онлайн-страхування, який надає можливість порівняти пропозиції і замовити страхові поліси від 30 найбільш відомих страхових компаній України з урахуванням знижок та бонусів.

На сайті є онлайн-калькулятор для розрахування наступних видів страховок:

- ОСАЦВ;
- КАСКО;
- зелена карта;
- туристичне страхування.

На рис. 2.5 можна побачити, як виглядає інтерфейс онлайн-калькулятора на сайті Polismart.com.ua для розрахування вартості полісу ОСАЦВ.

The interface shows the following fields and options:

- Тип ТС** (Vehicle Type): Легковой до 1 600 см3
- Город регистрации ТС** (City of registration): Киев
- Год выпуска ТС** (Year of release): 2012
- Льготы** (Discounts):
  - ☒ Льгот нет
  - ☐ Пенсионер
  - ☐ Участник войны
  - ☐ Инвалид II группы
  - ☐ Чернобылец I-й или II-й кат.
- Сфера использования** (Sphere of use):
  - ☒ Физлицо
  - ☐ Юрлицо
- Используется ли в такси** (Used as a taxi):
  - ☐ Да
  - ☒ Нет

A blue button labeled **РАССЧИТАТЬ** (Calculate) is located at the bottom right of the form.

Рисунок 2.5 – Интерфейс онлайн-калькулятора проекту Polismart для розрахування вартості полісу ОСАЦВ.

Для того, щоб скористатися калькулятором необхідно вказати:

- тип вашого транспортного засобу та об'єм двигуна;
- рік випуску;
- право на пільги;
- сферу використання;
- місце реєстрації;
- чи використовується в якості таксі.

Результат пошуку зображений на рис. 2.6.






































Компанія	Бонуси	Хар-ки	Франшиза	ДГО	Итого СК	Итого РМ	
 ★★★★★(0)	 ДОСТАВКА 30 ГРН ПО КИЕВУ ЗА 3 Ч	   	1000 грн	0 грн	<del>1296</del> грн	699 грн	КУПИТЬ Сравнить <input type="checkbox"/> ▼ Подробнее
 ★★★★★(0)	 ДОСТАВКА 30 ГРН ПО КИЕВУ ЗА 3 Ч	   	0 грн	0 грн	<del>1296</del> грн	737 грн	КУПИТЬ Сравнить <input type="checkbox"/> ▼ Подробнее
 ★★★★★(0)	 ДОСТАВКА 30 ГРН ПО КИЕВУ ЗА 3 Ч	   	0 грн	0 грн	<del>1296</del> грн	763 грн	КУПИТЬ Сравнить <input type="checkbox"/> ▼ Подробнее
 ★★★★★(0)	 ДОСТАВКА 30 ГРН ПО КИЕВУ ЗА 3 Ч	   	0 грн	0 грн	<del>1296</del> грн	881 грн	КУПИТЬ Сравнить <input type="checkbox"/> ▼ Подробнее
 ★★★★★(0)	 ДОСТАВКА 30 ГРН ПО КИЕВУ ЗА 3 Ч	   	0 грн	0 грн	<del>1296</del> грн	899 грн	КУПИТЬ Сравнить <input type="checkbox"/> ▼ Подробнее
 ★★★★★(0)	 ДОСТАВКА 30 ГРН ПО КИЕВУ ЗА 3 Ч	    	0 грн	0 грн	<del>1296</del> грн	915 грн	КУПИТЬ Сравнить <input type="checkbox"/> ▼ Подробнее

Рисунок 2.6 – Результат розрахування вартості полісу ОСАЦВ в онлайн-калькуляторі на сайті Polismart.com.ua.

Користувачу представлені наступна інформація:

- назва страхової компанії;
- бонусні пропозиції;
- Характеристики;
- Розмір франшизи;
- ДСВ (ДГО).

## 2.4 Висновки

Кожна з онлайн-страхових служб використовує певні критерії, щоб знайти найбільш точний страховий поліс. Більше того, чим вища кількість цих коефіцієнтів, тим біль релевантним буде результат пошуку, а це означає, що покупець з більшою ймовірністю здійснить замовлення на сайті.

Однак, як видно з таблиці. 2.1, жодна з існуючих служб (ОХ.ЮА, Фінсканер, Полісмайт) не надає користувачеві найбільш точний пошук, враховуючи всі деталі та особливості транспортного засобу та водія.

Таблиця 2.1 – Порівняння пошукових форм існуючих продуктів страхування-онлайн з розроблювано системою

	ОХ.ЮА	Фінсканер	Полісмарт	Розроблювана система
Вид авто	+	+	+	+
Об'єм см3	+	+	+	+
Реєстрація	+	+	+	+
Водійський строк	-	+	-	+
Рік авто	-	-	+	+
Наявність пільг	-	-	+	+
Авто в таксі	+	-	+	+
Страховий термін	-	-	-	+

Так, наприклад, Фінсканер здійснює пошук страхових полісів ОСАЦВ за типом транспортного засобу, об'ємом двигуна, стажем керування водія, але не враховує рік випуску авто та право водія на пільги. А от у формі пошуку сервісу Полісмарт є пункт про рік випуску автомобіля на відміну від Фінсканер, проте немає пункту про стаж керування водія.

Таким чином, система, створення якої описується в даній магістерській дисертації, покликана забезпечити виконання усі зазначених перерахованих вище проблем.

Крім того, в системі реалізований модуль – «Гараж онлайн», який складається з двох розділів:

– «Автоматична реєстрація витрат» – дозволяє контролювати свої витрати, контролювати їх, створювати автоматичні нагадування про майбутні ТО, та планувати графік їх проходження;

– «Власне Авто» – дозволяє зручно контролювати стан автомобіля, оновлювати пробіг і проводити на його планове технічне обслуговування. Система надсилає повідомлення користувачам, заохочує їх вчасно підключитися до послуги та оновити свої страхові поліси.

### 3 РОЗРОБЛЕННЯ СИСТЕМИ ОНЛАЙН-СТРАХУВАННЯ ТА ОБЛІКУ СЕРВІСНИХ РОБІТ АВТОТРАНСПОРТУ

В цьому розділі магістерської дисертації буде розглянута розробка [12] системи онлайн страхування та обліку сервісних робіт транспортних засобів під назвою «Fa-Fa Club».

В результаті аналізу ринку аналогічних рішень був обраний варіант створення саме такої системи, яка відрізнятиметься від своїх конкурентів розширеним функціоналом, більш детальним пошуком по страховим компаніям та полісам, а також унікальним сервісом – «Віртуальний гараж», що дозволить зібрати всю інформацію про авто і необхідні послуги в одному місці.

Для розробки «Fa-Fa Club» будуть використані наступні технології, програмні середовища та платформи, як: Ubuntu, Apache2, PHP-фреймворк Laravel, JavaScript [9], jQuery [6], Bootstrap, HTML5/CSS3 [10].

#### 3.1 Формули для обчислення вартості страхових полісів системи

Розрахунок вартості страхових полісів в системі здійснюється з урахуванням показників, визначених національним законодавством.

Так, відповідний закон [2] встановлює наступну схему розрахунку:

Вартість поліса = Базова вартість • K1 (тип авто) • K2 (регіон реєстрації) • K3 (сфера використання) • K4 (стаж) • K5 (період страхування) • K6 (використання в таксі) • K7-«строк» • K8-«Бонус-Малус» • K9 (чи є знижка пільговику).

Базова вартість становить 180 гривень за законом [2].

K1 (автомобільний). Машини поділяються на 12 типів - від мотоциклів і скутерів з об'ємом до 250 см до бусів понад двадцятьма місцями і автомобілів з двигунами від 3000 см<sup>3</sup>.

K2 (регіон обліку транспортного засобу). Чим більший населений пункт, тим більше шансів, оскільки ймовірність аварії в великому місті та віддаленому селі дуже



різна. Наприклад, у Харкові коефіцієнт помножується на 5,20, а для малих міст і сіл на 2,30.

K3 (де буде використовуватись автомобіля). Вартість поліса також залежить від сфери використання автомобіля. Фізичні особи мають коефіцієнт від 0,15 до 2,30, а ТОВ від 0,25 до 2,50.

K4 (досвід водіння). Встановлюється від 1.21 до 1.51 враховуючи автомобільний досвід водія.

K5 (термін дії контракту). Мінімальний термін реєстрації складає пів року.

K6 (таксі). Якщо автомобіль використовується як таксі, коефіцієнт становить 2,0

K7 (хронометраж). Якщо договір обов'язкового страхування укладається менше ніж на півроку, то впливає на вартість поліса.

K8 (Bonus-malus) [4]. Якщо протягом року не мав ДТП з власної вини, він має законне право на знижку 5% протягом цього періоду.

K9 (спеціальний дисконт). Ветерани, інваліди, пенсіонери тощо.

Особливістю автомобільного страхування є те, що досить часто, і для задоволення всіх, стягується премія, яка значною мірою залежить від претензій, поданих у полісі в минулому. У таких системах рейтингу досвіду бонуси можна отримати, не подаючи претензій, а малус виникає, коли подано багато претензій.

Системи оцінювання досвіду є загальноприйнятою практикою перестрахування, але в цьому випадку це впливає безпосередньо на споживача. Насправді, у випадку випадкового коливання премії, кінцева мета страхування, тобто перебування у повністю надійному фінансовому становищі щодо цього конкретного ризику, не досягається.

коли частина вимог не відшкодовується страховиком, оскільки існує франшиза.

### 3.2 Розробка системи

Fa-Fa Club - це система, яка реалізована для підбору та оформлення [11] страхових продуктів, замовлення послуг, пов'язаних з використанням власного транспортного засобу, онлайн запису на технічне обслуговування, відображення цін

на паливо тощо. Також сервіс надає можливість консолідації інформації з використання автомобіля і його обслуговування (бортовий журнал, планування графіку ТО, нагадування про необхідні сервісні роботи, облік витрат). Весь функціонал доступний в режимі онлайн на сайті, що дозволяє кінцевому користувачеві мати завжди актуальні дані під рукою і здійснювати будь-які операції в кілька кроків.

### 3.2.1 Модуль підбору страхової компанії та купівлі полісу

Модуль підбору страхової компанії та купівлі полісу призначений для того, щоб потенційний клієнт (власник автотранспортного засобу) міг легко та зручно без зайвих зусиль обрати страхову компанію та придбати у неї страховий поліс для свого авто не виходячи з дому.

В системі «Fa-Fa Club» модуль має назву «Калькулятор страховки» і передбачає розрахунок страховки за певними параметрами, які вказує користувач. Для кожного виду страхування необхідно вказувати різні параметри.

Так, для підбору страхового полісу ОСАЦВ (обов'язкового страхування цивільної відповідальності автовласників) необхідно вказати наступні дані (рис. 3.1, рис. 3.2):

- тип транспорту (легковий, вантажний автомобіль, мотоцикл, автобус, причеп);
- об'єм двигуна – для легкових автомобілів та мотоциклів, вантажопідйомність.

**КАЛЬКУЛЯТОР СТРАХОВКИ**

Рассчитайте стоимость страховки по параметрам

ОСАГО
ЗЕЛЕНАЯ КАРТА

**Тип транспорта**






 Легковые автомобили	 Грузовые автомобили	 Мотоциклы	 Автобусы	 Прицепы
Объем до 1600см <sup>3</sup> включительно	Объем 1601-2000см <sup>3</sup> включительно	Объем 2001-3000см <sup>3</sup> включительно	Объем больше 3000см <sup>3</sup>	

Рисунок 3.1 – Калькулятор страховки системи «Fa-Fa Club» для ОСАЦВ

- місце реєстрації;
- чи використовується в якості таксі;
- стаж водіння;
- рік випуску транспортного засобу;
- вид пільг, якщо вони є;
- бажаний період дії страхового полісу.

Рисунок 3.2 – Калькулятор страховки системи «Fa-Fa Club» для ОСАЦВ

Крім розрахунку вартості страховки ОСАЦВ наведеного у Додатку Г, модуль підбору страхової компанії та купівлі полісу надає також можливість розрахувати вартість та придбати онлайн поліс страхування «green card».

«Green Card» – це міжнародно визнаний страховий документ, який надають правоохоронним органам для підтвердження страхового покриття. Поліція в країні, яку відвідую власник карти, засвідчує, що автомобіліст має мінімально необхідну страхове страховку.

Для того, щоб розрахувати вартість полісу «зелена карта» наведено у Додатку Д, та придбати його в системі «Fa-Fa Club», необхідно в страховому калькуляторі вказати наступні дані (рис. 3.3, рис. 3.4):

- тип транспорту (легковий, вантажний автомобіль, мотоцикл, автобус, причеп);
- країни, які передбачаються для відвідування із полісом (країни Європи або Росія, Азербайджан, Білорусь, Молдова);

- бажаний період дії поліса (від 15 днів до 1 року);
- тип транспортного засобу
- право власності
- наявність декількох транспортних засобів

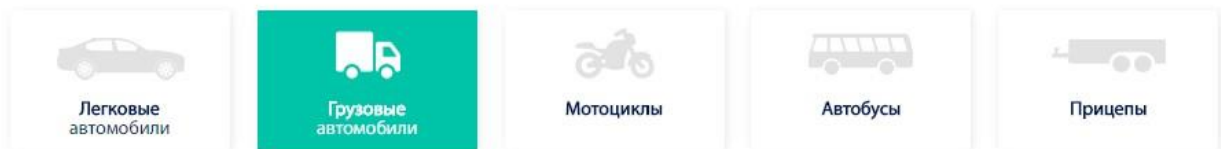
## КАЛЬКУЛЯТОР СТРАХОВКИ

Рассчитайте стоимость страховки по параметрам

ОСАГО

ЗЕЛЕНАЯ КАРТА

### Тип транспорта



- Рисунок 3.3 – Калькулятор страховки системы «Fa-Fa Club» для «зеленой карты»

### Срок действия полиса

15 дней	1 месяц	2 месяца	3 месяца	4 месяца
5 месяцев	6 месяцев	7 месяцев	8 месяцев	9 месяцев
10 месяцев	11 месяцев	1 год		

### Страны для посещения

- ☐ Азербайджан, Беларусь, Молдова и Россия
- ☒ Страны Европы (Австрия, Македония, Албания, Марокко, Азербайджан, Нидерланды, Бельгия, Норвегия, Болгария, Польша, Босния и Герцеговина, Португалия, Великобритания, Венгрия, Румыния, Германия, Сербия, Греция, Словакия, Дания, Словения, Израиль, Тунис, Иран, Турция, Ирландия, Финляндия, Исландия, Франция, Испания, Хорватия, Италия, Черногория, Кипр, Чехия, Латвия, Швейцария, Литва, Швеция, Лихтенштейн, Эстония, Люксембург, Мальта)





Сравнить цены

Рисунок 3.4 – Калькулятор страховки системы «Fa-Fa Club» для «зеленой карты»

Як має виглядати результат пошуку [13] страхового полісу за вказаними критеріями зображено на рис. 3.5. Відображені поліси можна відсортувати за вартістю, за назвою страхової компанії, що пропонує поліс, та за франшизою.

## РЕЗУЛЬТАТЫ ПОИСКА ПОЛИС «ОСАГО»

По Вашим критериям найдено 10 результатов

Логотип	Полис ОСАГО	Цена	Цена со скидкой	Франшиза	Скидка	Кнопка
	Полис ОСАГО Компания: СК "Unives" <a href="#">Подробнее</a>	1477 грн	886 грн	1000 грн	40 %	Купить
	Полис ОСАГО Компания: СК "Unives" <a href="#">Подробнее</a>	1556 грн	934 грн	500 грн	40 %	Купить
	Полис ОСАГО Компания: СК "VUSO" <a href="#">Подробнее</a>	1477 грн	946 грн	1000 грн	36 %	Купить
	Полис ОСАГО Компания: СК "Unives" <a href="#">Подробнее</a>	1675 грн	875 грн	0 грн	40 %	Купить

**Отображать:**

**Цена**

☒ По возрастанию

☐ По убыванию

**Страховая компания**

☐ По алфавиту

**Франшиза**

☐ По возрастанию

☐ По убыванию

**Новый поиск**

Рисунок 3.5 – Результат розрахування вартості полісу ОСАЦВ в онлайн-калькуляторі системи «Fa-Fa Club»

Якщо клієнта влаштовуватимуть запропоновані варіанти, то він матиме змогу одразу придбати поліс онлайн, алгоритм замовлення наведено в Додатку В, та замовити доставку кур'єром або ж самостійно забрати поліс із офісу «Fa-Fa Club». Форма для замовлення та оплати зображена на рис. 3.6.

Хоча окремі агенції, брокери або моделі корпоративних агентств під керівництвом банку перебувають у режимі офлайн і потребують особистих відносин, повністю онлайн-модель також існує у цьому просторі вже досить багато років. І ці електронні страхові компанії відрізняються від традиційних страховиків. «Компанії електронного страхування працюють за прямою моделлю для споживачів через Інтернет-канали, щоб продавати поліси та обслуговувати клієнтів без фізичних агентів та брокерів, що вступають у суперечку

**Шаг 1** Шаг 2 Шаг 3 Оплата / Доставка

**Личная информация**

Имя \* Сергей

Отчество \* Ваше отчество

Фамилия \* Иванов

Дата Рождения \* ДД-ММ-ГГГГ

Номер телефона \* +380 (09) 645 87 98

E-Mail \* Ваш E-Mail

**Оформить по телефону** **Далее**

☐ Я предоставляю свое согласие на передачу, сбор, обработку, использование, накопление и хранение моих персональных данных с целью их внесения в базу персональных данных контрагентов согласно норм действующего [законодательства о защите персональных данных](#).

**Детали покупки**

ОСАГО.  
Компания: СК «АХА  
Страхование»  
Отзывов: 0

Цена  
**1330 грн**

Франшиза  
**0 грн**

**Итого**

Стоимость: 1330 грн  
Доставка: 0 грн  
**К оплате: 1330 грн**

Рисунок 3.6 – Форма для замовлення та оплати страхового полісу через систему «Fa-Fa Club»

### 3.2.2 Модуль обліку сервісних робіт та витрат автотранспорту

Модуль обліку сервісних робіт та витрат автотранспорту [14] має назву «Гараж онлайн» в системі «Fa-Fa Club». Це унікальний сервіс і на сьогоднішній день в Україні він не має аналогів.

«Гараж онлайн» дозволяє зібрати всю інформацію про автомобіль та необхідні послуги в одному сховищі. Це зручний сервіс для користувачів транспортних засобів по веденню корисної статистики використання автотранспорту (витрати на паливо, витрати на ТО, ремонт тощо), формування графіків технічного контролю та багато чого іншого.

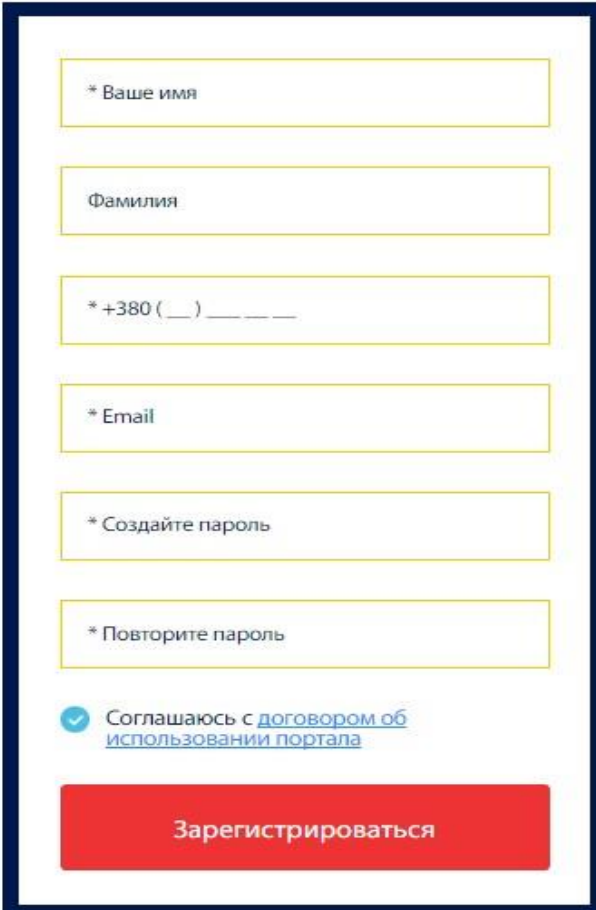
Після реєстрації користувач в особистому кабінеті також має доступ до списку сервісних точок (СТО, шиномонтаж, автомийки), що здійснюють послуги за знижками для клієнтів «Fa-Fa Club», з адресами, маршрутами і можливістю бронювання. Бронювання, замовлення і оплата послуг здійснюються у декілька кліків.

Щоб отримати доступ до модулю «Гараж онлайн» та його функціоналу, необхідно зареєструватися в системі «Fa-Fa Club». Форма реєстрації проста і має мінімальну кількість обов’язкових полів для швидкої реєстрації – прізвище клієнта, ім’я, номер телефону, електронна пошта, пароль.

Онлайн-реєстрація та авторизація OAuth повністю інтегрована у базу даних. Це означає, що кожен зареєстрований користувач або вже матиме профіль у базі даних, або буде він буде створений під час реєстрації.

Залежно від самого тип аккаунту інформація про нього зберігається у таблиці адміністратори чи користувачі.

Форма реєстрації нового користувача в системі «Fa-Fa Club» зображена на рис. 3.7.



\* Ваше имя

Фамилия

\*+380 ( \_\_ ) \_\_\_\_

\* Email

\* Создайте пароль

\* Повторите пароль

☒ Соглашаюсь с [договором об использовании портала](#)

Зарегистрироваться

Рисунок 3.7 – Форма реєстрації нового користувача в системі «Fa-Fa Club»



Після реєстрації користувач отримує повний доступ до функціоналу модуля обліку сервісних робіт та витрат автотранспорту («Гараж онлайн»), який має наступні розділи:

1. «Бортовой журнал»;
2. «Автомобиль»;
3. «Расходы»;
4. «Страховка».

У розділі «Бортовой журнал» здійснюється контроль за всіма сервісними роботами, які необхідно здійснити з автотранспортом (пройти ТО, здійснити шиномонтаж, змінити мастило у двигуні тощо). Інтерфейс розділу «Бортовой журнал» зображений на рис. 3.8.

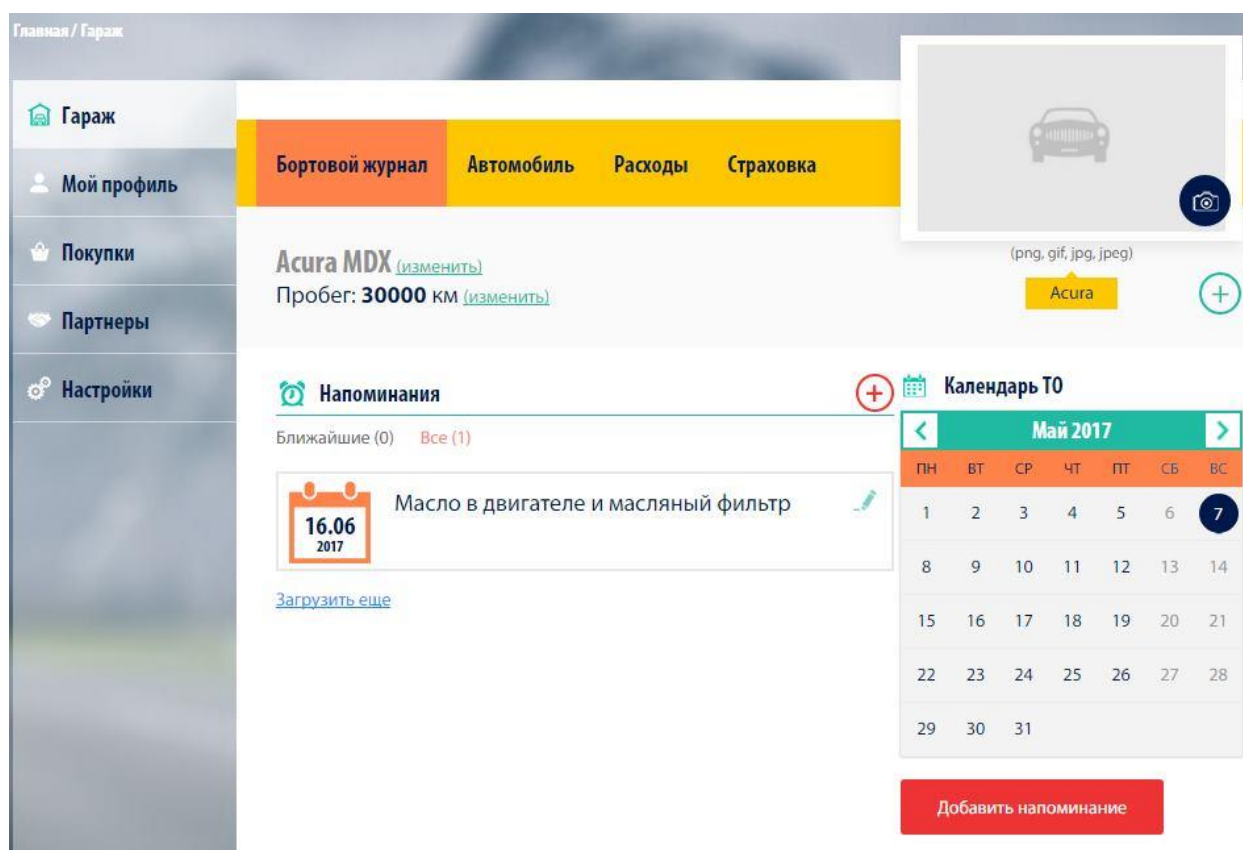


Рисунок 3.8 – Розділ «Бортовой журнал» модуля обліку сервісних робіт системи «Fa-Fa Club»

У цьому ж розділі користувач має можливість додавати нагадування про необхідність проходження регулярного обов’язкового технічного контролю (якщо такий необхідний для його типу транспортного засобу) або виконання тих чи інших



сервісних робіт. Інтерфейс для додавання нового нагадування зображений на рис. 3.9.

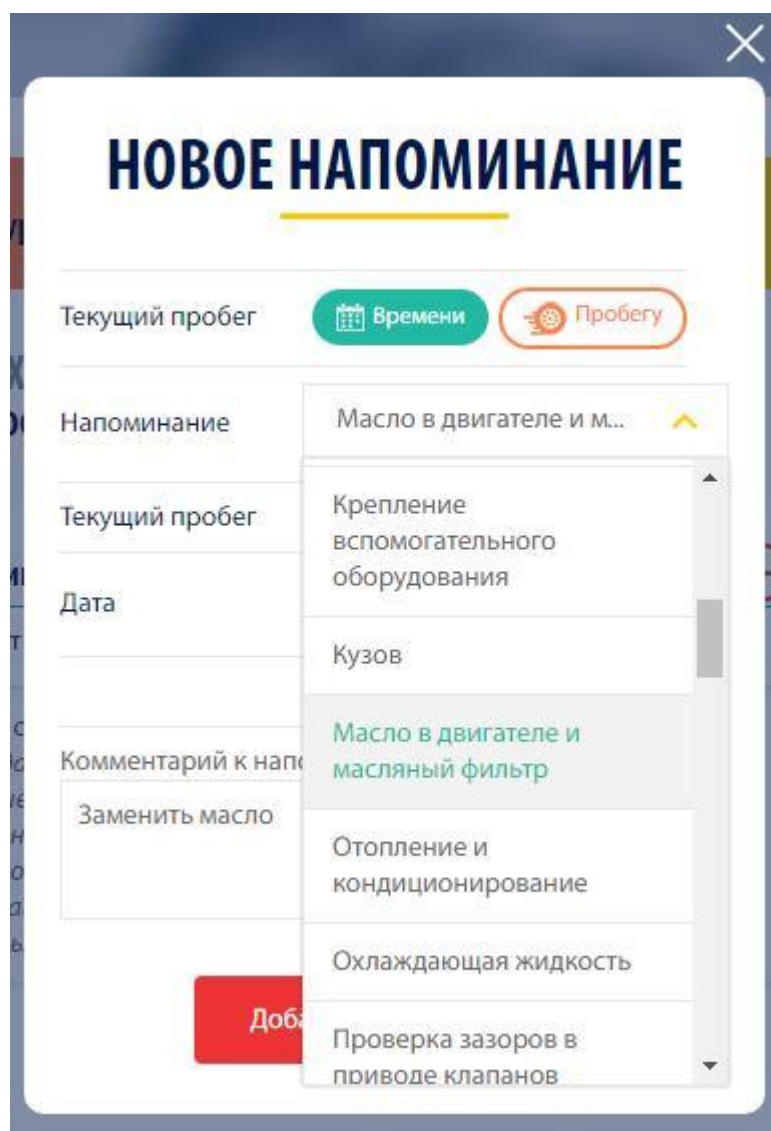


Рисунок 3.9 – Додавання нового нагадування у модулі обліку сервісних робіт системи «Fa-Fa Club»

Розділ «Автомобиль» призначений для того, щоб користувач міг зручно стежити за станом свого автотранспорту, оновлювати пробіг і виконувати необхідні регламентні роботи на його основі. Інтерфейс розділу «Автомобіль» зображений на рис. 3.10.

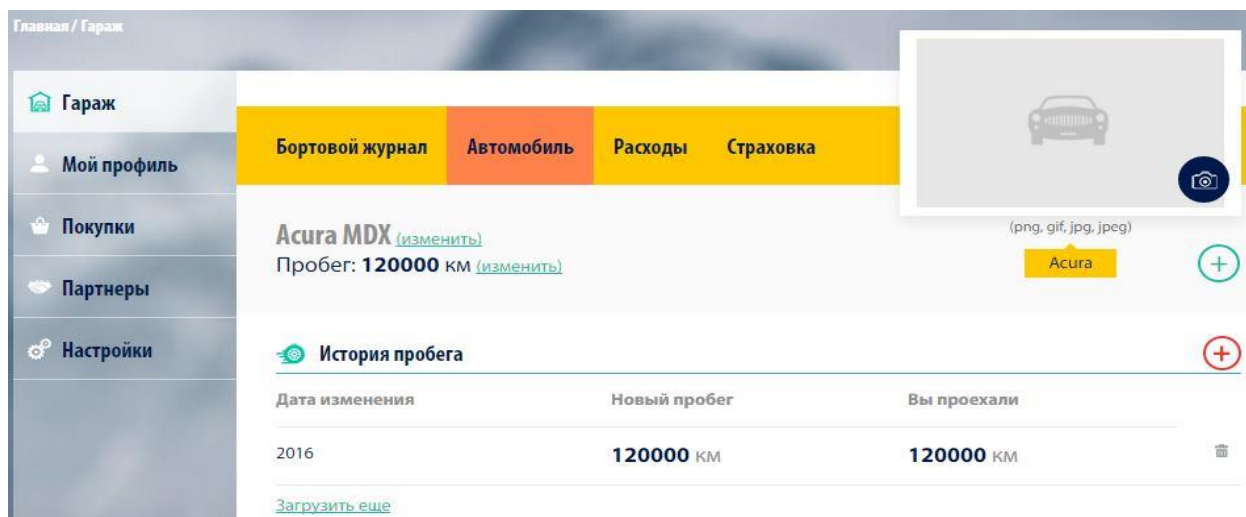
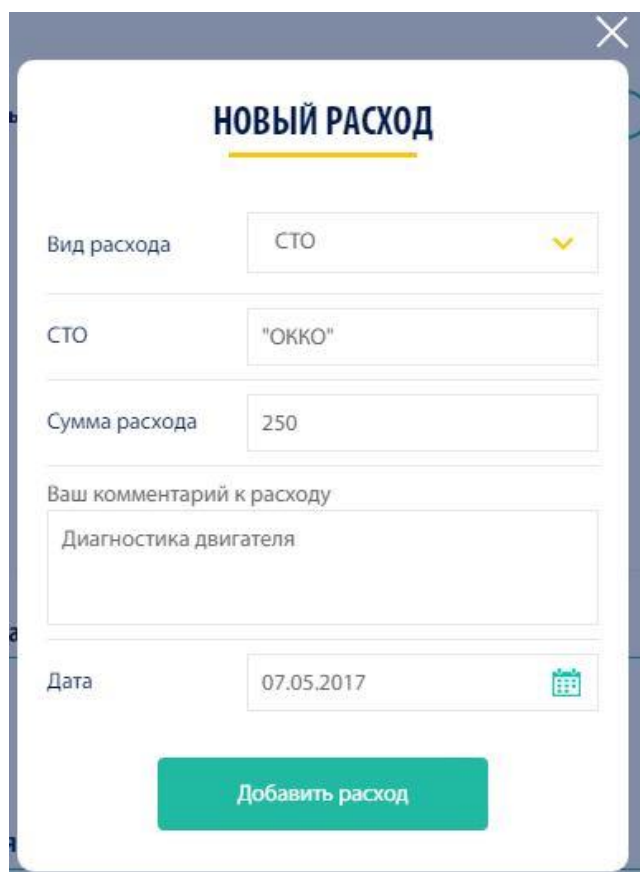


Рисунок 3.10 – Розділ «Автомобиль» модуля обліку сервісних робіт системи «Fa-Fa Club»

Крім того, саме у цьому розділі здійснюється додавання автотранспорту, для роботи із яким, власне, і призначена система «Fa-Fa Club». У формі для додавання авто обов'язковими для заповнення є поля марка та модель. Також, за бажанням, користувач може зазначити рік випуску, об'єм двигуна, номер кузова (VIN), актуальний пробіг. Пізніше користувач може додати також фото свого транспортного засобу. Інтерфейс для додавання автомобіля зображений на рис. 3.11.

Рисунок 3.11 – Додавання нового автомобіля у модулі обліку сервісних робіт системи «Fa-Fa Club»

Розділ «Расходы» призначений для відстеження всіх коштів, витрачених на авто. Користувач самостійно додає витрати, вказуючи вид витрати, назву сервісної точки, де була здійснена витрата, суму, дату та коментар, якщо необхідно. Обов'язковими полями є вид, сума та дата витрати. Інтерфейс для додавання нової витрати зображений на рис. 3.12.



Скриншот інтерфейсу для додавання нової витрати. Формування заголовку «НОВЫЙ РАСХОД» підкреслено жовтою лінією. Поля форми:

- Вид расхода: СТО (з вибором)
- СТО: "ОККО"
- Сумма расхода: 250
- Ваш комментарий к расходу: Диагностика двигателя
- Дата: 07.05.2017 (з календарем)

Нижче розташований зелений кнопка «Добавить расход».

Рисунок 3.11 – Додавання нової витрати у модулі обліку сервісних робіт системи «Fa-Fa Club»

Крім того, у розділі «Расходы» є можливість здійснювати контроль за витратами за допомогою графіків та таблиць.

На графіках інформація про витрачені на автотранспорт кошти відображається за типом витрат – запчастини та аксесуари, мийка та хімчистка, страхування, шиномонтаж, паливо, інше. Відповідно, на графіку можна побачити лише тип витрат та загальну суму, витрачену саме на цей тип (рис. 3.12).

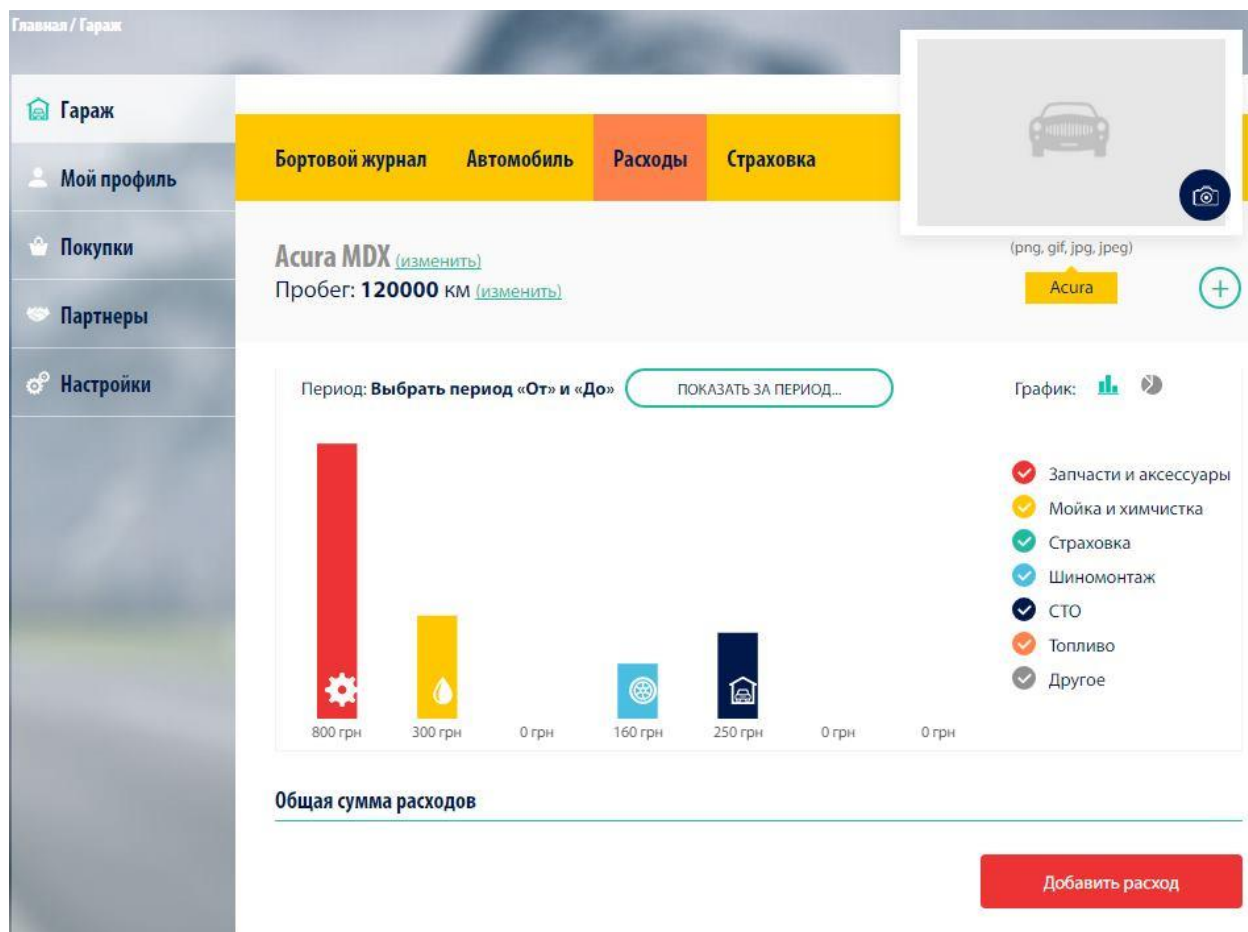


Рисунок 3.12 – Розділ «Расходы» модуля обліку сервісних робіт системи «Fa-Club». Відображення витрат у вигляді графіка

Для більш детального ознайомлення з витратами за певний період зручно використовувати таблиці (рис. 3.13), які також доступні у даному розділі.

Інтерфейс дозволяє побачити скільки було витрачено коштів на:

- запчастини та аксесуари;
- мийка та хімчистка;
- страховка;
- шиномонтаж;
- СТО

История расходов

Все

СТО

Топливо

Запчасти

Мойка

Шиномонтаж

Страховка

Другие

28.04

2017

СТО

-250 грн

07.03

2017

Шиномонтаж

Летняя резина

-160 грн

15.02

2017

Мойка и химчистка

Комплекс

-300 грн

11.01

2017

Запчасти и аксессуары

Замена лампочек в фарах

-800 грн

Рисунок 3.13 – Розділ «Расходы» модуля обліку сервісних робіт системи «Fa-Fa Club». Відображення витрат у вигляді таблиці

У розділі «Страховка» власник транспортного засобу додає свою актуальну страховку, вказуючи строк її дії, щоб система «Fa-Fa Club» попередила його про закінчення строку дії поліса та нагадала про необхідність придбання нового. Одразу ж у цьому розділі користувач може перейти до онлайн калькулятора, щоб розрахувати вартість нового страхового полісу та обрати найбільш вигідний варіант. Інтерфейс розділу «Бортовой журнал» зображений на рис. 3.14.

Калькулятор автомобільного страхування надає клієнтам багато варіацій щодо рівня премій відповідно до їх вимог. Для розрахунку точної страхової премії для автомобіля клієнт повинен вказати конкретні дані, які включають дату реєстрації, дані про автомобіль, дату початку полісу та інші додаткові деталі. Це лише кілька кліків мишею, і клієнт отримає вигідну страховку.

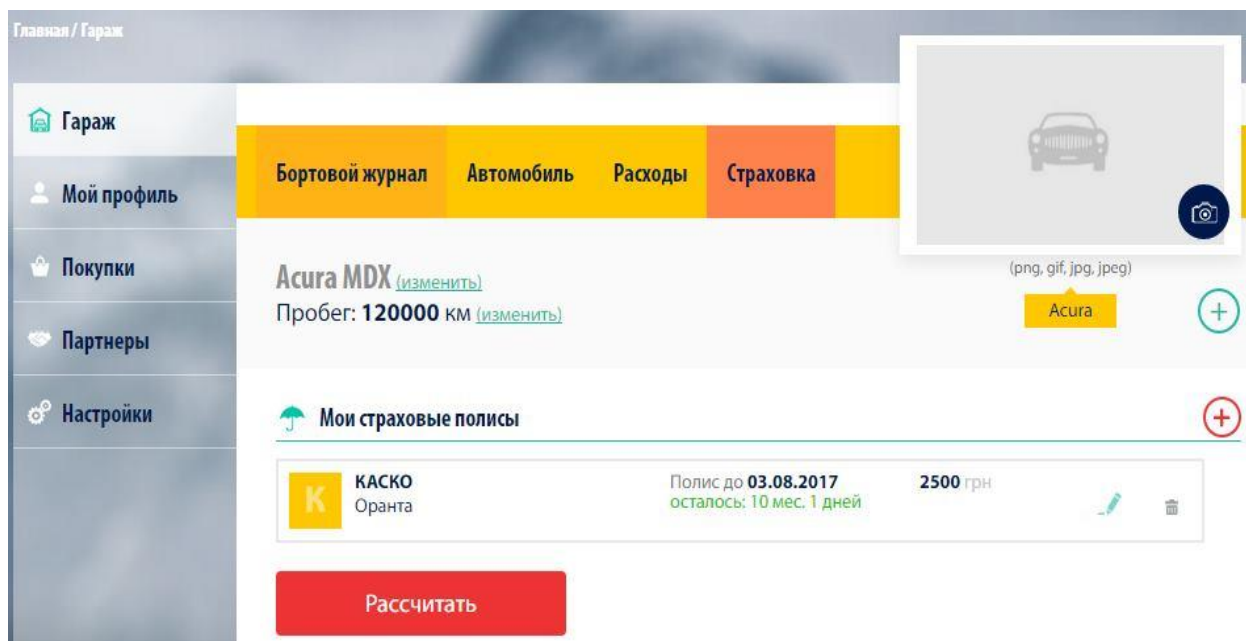


Рисунок 3.14 – Розділ «Страховка» модуля обліку сервісних робіт системи «Fa-Fa Club»

Крім усіх вищезазначених функцій, модуль обліку сервісних робіт та витрат «Гараж онлайн» надає також доступ до списку сервісних точок, які надають послуги або продають деталі та аксесуари для автотранспорту за спеціальними знижками для клієнтів системи «Fa-Fa Club». Для більш зручного пошуку такі сервісні точки можна відсортувати за категоріями (СТО, АЗС, автомийки, шиномонтаж, прокат авто, автосалони), а також за рейтингом та відгуками. Одразу на сайті системи «Fa-Fa Club» можна здійснити онлайн бронювання певної послуги/товару у обраній сервісній точці. Даний підрозділ має назву «Партнеры», його інтерфейс зображений на рис. 3.15.

Щоб розрахувати преміальні котирування для вживаних автомобілів, нам потрібно вказати такі дані, як тип автомобіля, тип палива, реєстраційний номер, дані існуючого полісу страхового автомобіля, зміна даних про право власності на автомобіль, якщо такі є, та звіти про позови попереднього року. Калькулятор для вживаних автомобілів дозволяє покупцям отримати найкращу угоду відповідно до вимог.

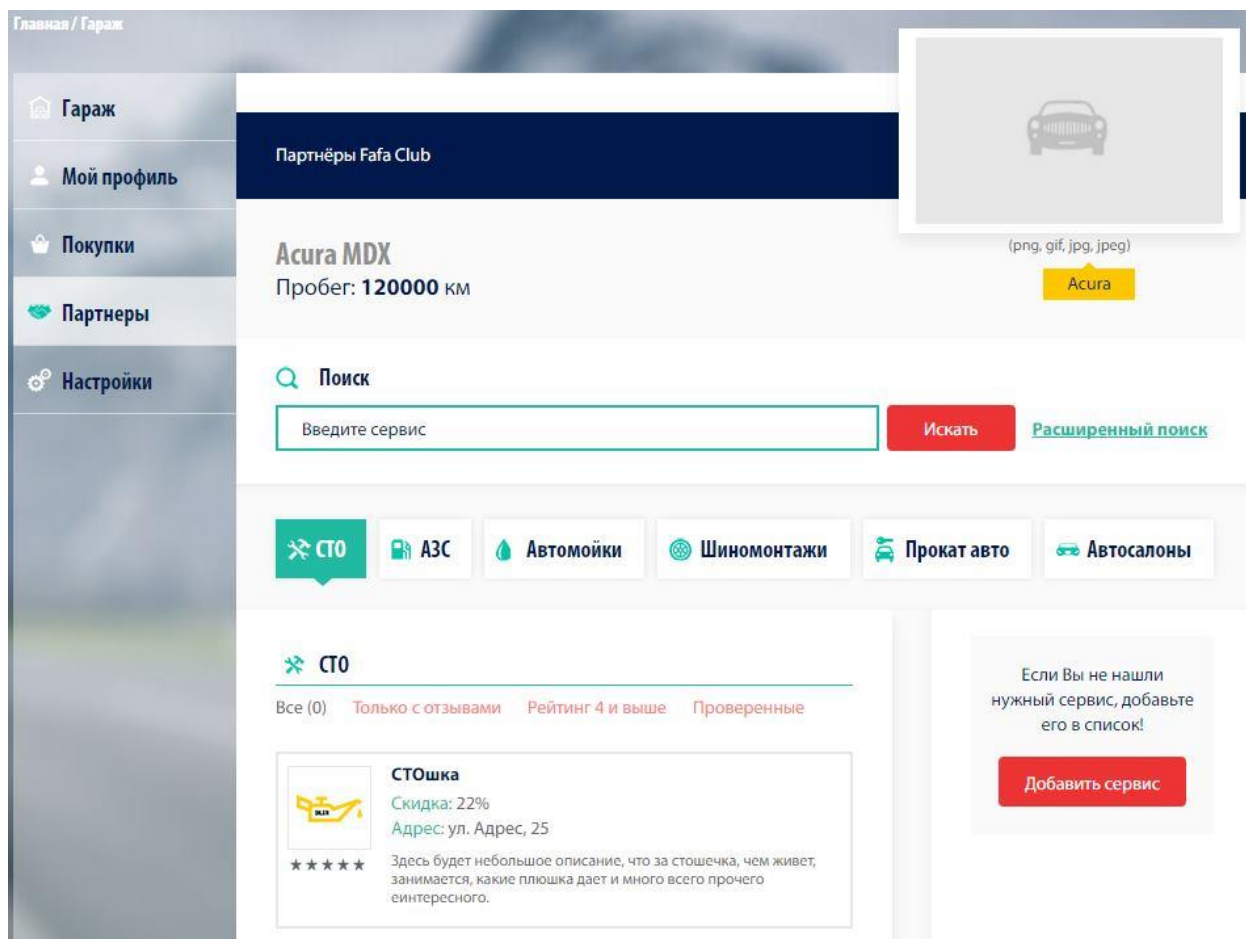


Рисунок 3.14 – Підрозділ «Партнёры» модуля обліку сервісних робіт системи «Fa-Fa Club»

В вище описаному модулі знаходяться підрозділи: «СТО», «АЗС», «Автомойка», «Шиномонтаж», «Прокат авто», «Автосалоны». Кожен з цих вкладок відображає перелік партнерів згрупованих по категоріям.

Користувач має змогу відкрити будь яку релевантну для нього вкладку та отримати інформацію по кожному з представлених в ній партнерів.

По кожному з партнерів надана наступна інформація:

- Назва організації
- Фірмовий логотип
- Розмір знижки
- Адреса
- Короткий опис
- Рейтинг партнера в системі



### 3.3 Встановлення програмного забезпечення та налаштування середовища розробки

#### 3.3.1 Встановлення та налаштування платформи віртуалізації Oracle VM VirtualBox

Virtual Box – програма абсолютно безкоштовна. Вперше була надана компанією InnoTek в 2007 р. в двох варіантах – з відкритим і закритим вихідними кодами, причому обидві були безкоштовні за умови некомерційного використання. У 2008 р. платформа перекуплена компанією Sun Microsystems, яка і займається її розробкою в даний час.

Платформа являє собою систему віртуалізації для host-систем Windows, Linux і Mac OS і забезпечує взаємодію з гостьовими операційними системами Windows (2000/XP/2003/Vista/7/10), Linux (Ubuntu / Debian / OpenSUSE / Mandriva і пр.), OpenBSD, FreeBSD, OS / 2 Warp. Більш детальне ознайомлення з платформою Oracle VM VirtualBox можна на сайті виробника <https://www.virtualbox.org/>.

Завантажити платформу, яка підходить під систему, можна за посиланням: <http://www.virtualbox.org/wiki/Downloads>

В рамках даної магістерської дисертації розглянемо роботу з платформою VirtualBox версії 6.0.24, її встановлення і налаштування в середовищі операційної системи Windows 10.

Після того, як інсталяційний пакет записаний на жорсткий диск, можна приступати до установки програми.

Після запуску інсталятора ми побачимо вікно. Натискаємо кнопку «Next» і в новому вікні погоджуємося з умовами ліцензійної угоди, поставивши прапорець «I accept the terms in the License Agreement». У наступному вікні (рис. 3.15) буде запропоновано вибрати компоненти для установки і задати розташування виконуваних файлів. За замовчуванням, усі компоненти встановлюються на жорсткий диск (а нам потрібні всі), а сама програма встановлюється в папку «Program Files» на системному диску.



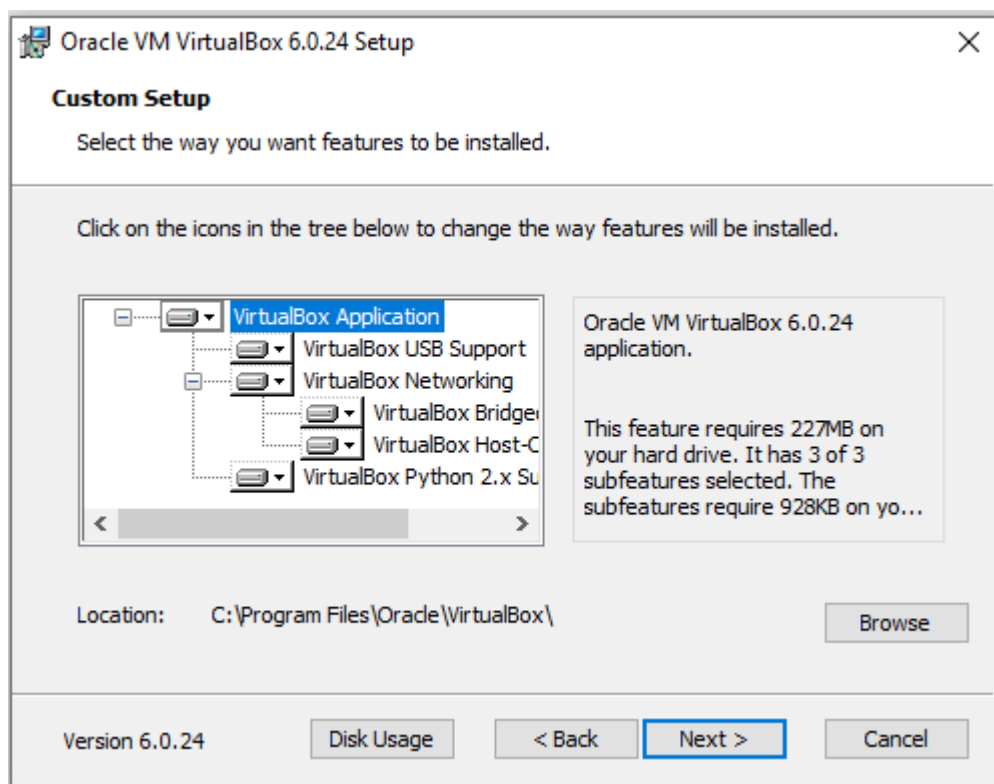


Рисунок 3.15 – Хід установки

Далі процес установки не потребує ніяких втручань, окрім дозволу на створення ярликів на робочому столі та в меню «Пуск». По закінченню установки програма повинна запуститися автоматично.

Інсталятор створить групу Oracle VM VirtualBox у меню «Пуск» Windows, що дозволить запустити програму та отримати доступ до її документації.

За стандартних налаштувань Oracle VM VirtualBox буде встановлений для всіх користувачів локальної системи. Якщо цього не потрібно, треба викликати програму встановлення, попередньо викликавши її наступним чином:

`VirtualBox.exe -extract`

Потім запустимо будь-яку з наведених команд у отриманому файлі .MSI. Це встановить Oracle VM VirtualBox лише для поточного користувача.

### 3.3.2 Встановлення Ubuntu версії 20.04

Убунту – це ([Ubuntu:];– людяність; «Убунту») –Linux, заснований на Дебіан GNU / Linux. Основним розробником і спонсором є компанія Canonical.

В даний час проект активно розвивається і підтримується вільним спільнотою.

За твердженнями Canonical, Ubuntu використовується приблизно 20 мільйонами користувачів по всьому світу . Він є 1-м в списку найбільш популярних дистрибутивів Linux для веб-серверів. За кількістю користувачів, які відвідали сайт виробника, займає 4-е місце .

Зазвичай нові версії дистрибутива виходять кожні півроку і підтримуються оновленнями безпеки протягом 9 місяців (починаючи з версії 13.04, до цього підтримка здійснювалася протягом півтора років).

Версії LTS, що випускаються раз в 2 роки, підтримуються протягом 5 років – як серверні, так і десктопні варіанти.

Для того, щоб встановити Ubuntu на свій комп'ютер, потрібно виконати наступні дії:

1. Завантажити образ диска Ubuntu (рис. 3.16) з офіційного сайту Ubuntu.

<https://ubuntu.com/download/>

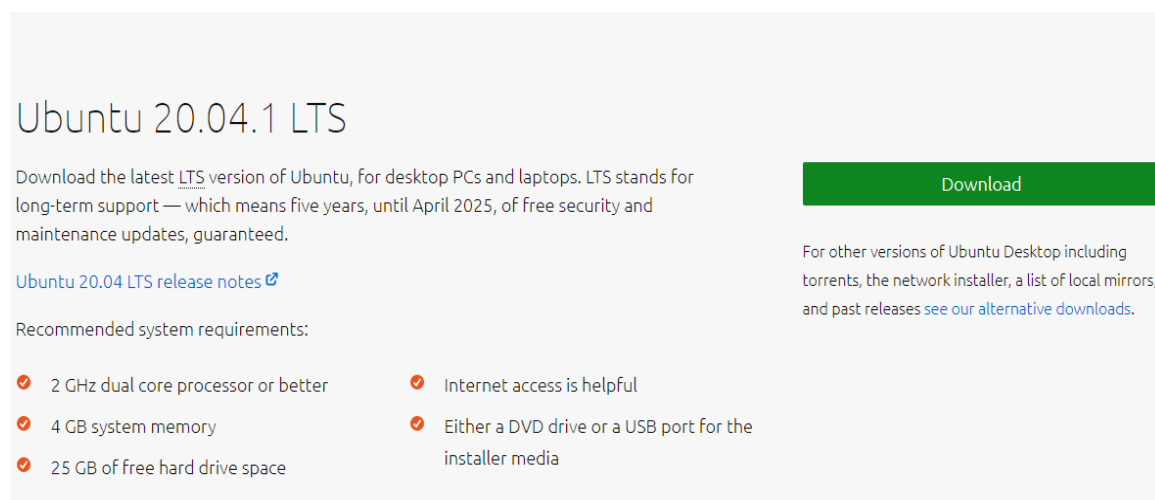


Рисунок 3.16 – Вікно завантаження Ubuntu

2. Запустивши VirtualBox VM та натиснувши «Machine» – «New» (рис. 3.17), далі потрібно ввести нову назву операційної системи у вікні підказки. Під час вводу імені, VirtualBox автоматично визначатиме тип і версію ОС. Однак її також можна змінити пізніше.

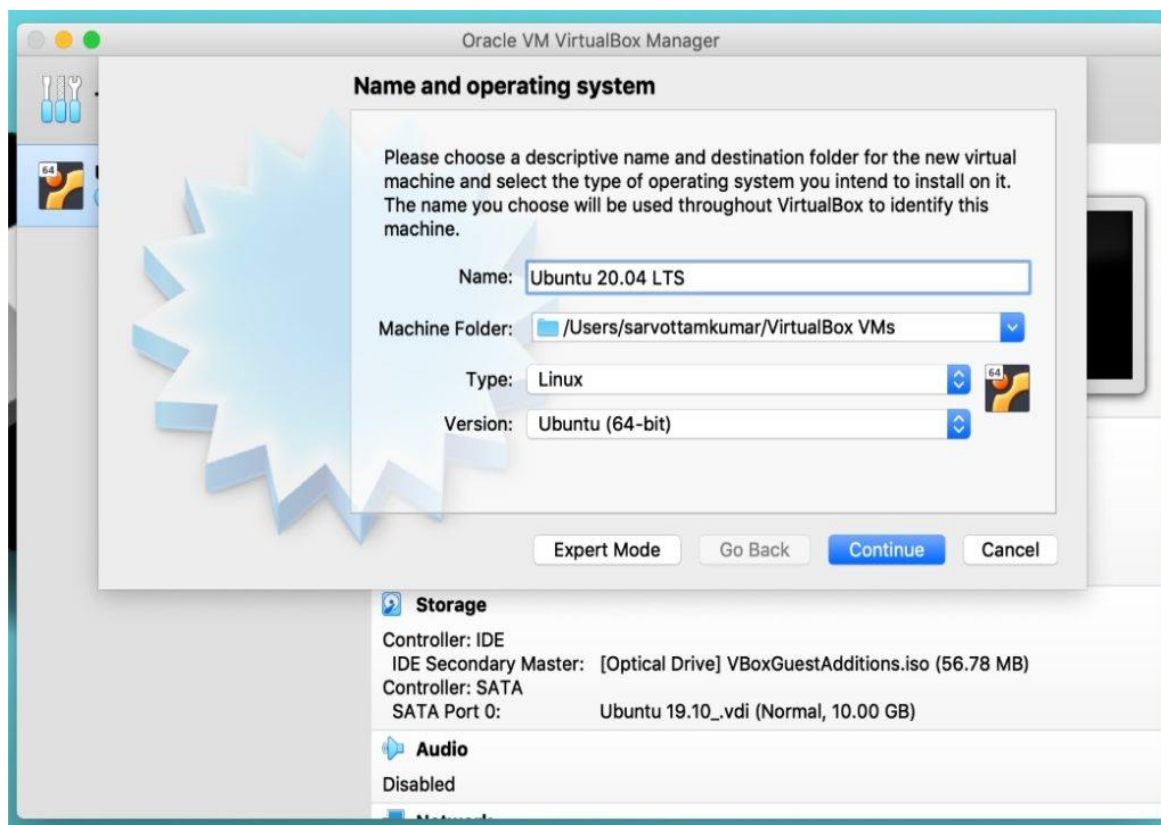


Рисунок 3.17 – Вікно створення нового віртуального середовища

Система перелаштовує код гостьової ОС, який зазвичай виконується в кільці 0, для виконання в кільці 1 на хості апаратного забезпечення. Оскільки цей код містить безліч привілейованих інструкцій, які не можуть виконуватися безпосередньо в кільці 1, VirtualBox використовує диспетчер сканування та аналізу коду (CSAM) для рекурсивного сканування коду кільця 0 перед першим виконанням для виявлення проблемних інструкцій, а потім викликає диспетчер виправлень (PATM) для проведення виправлення на місці. Це замінює інструкцію переходом до еквівалентного компільованого фрагменту коду в пам'яті гіпервізора, безпечного для VM.

3. Натиснувши продовжити та перейшовши до наступного кроку (рис. 3.18), щоб вибрати розмір оперативної пам'яті. Тут слід вибрати базову пам'ять, виходячи з обсягу оперативної пам'яті системи. Наприклад, якщо у хост-системі є 8 ГБ або 4 ГБ оперативної пам'яті, слід вибрати половину (4 ГБ або 2 ГБ) для кращої роботи. Однак настійно рекомендується не вибирати менше 1 ГБ.

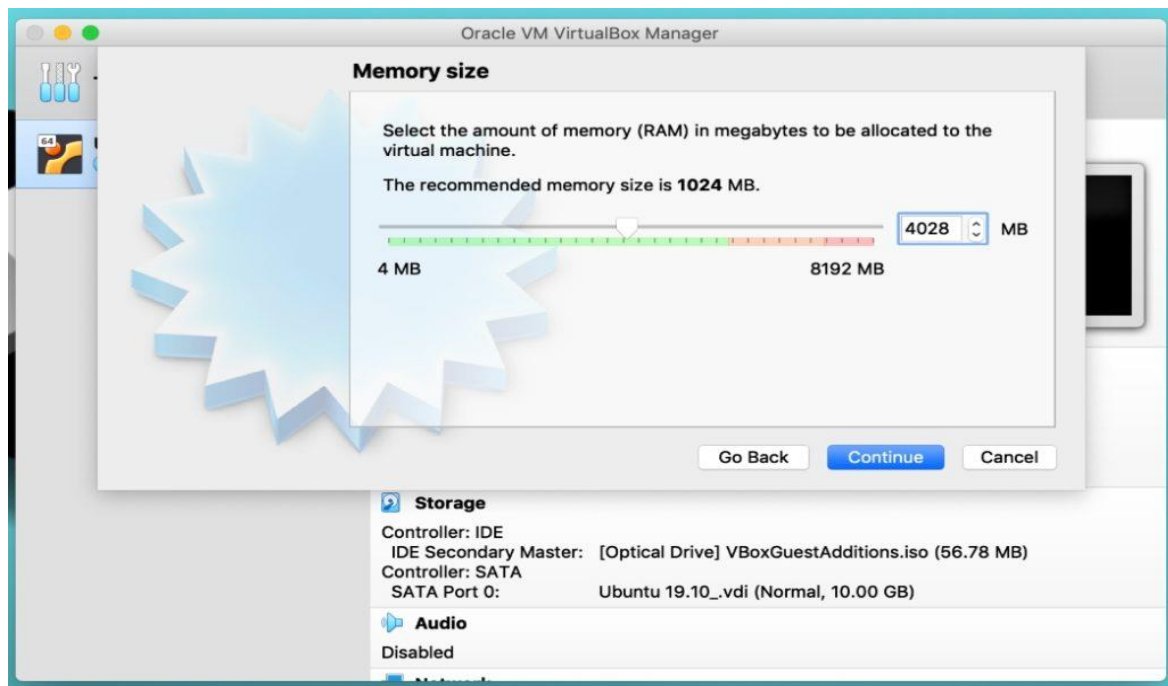


Рисунок 3.18 – Вікно створення нового віртуального середовища

4. Тепер потрібно виділити місце для віртуальних жорстких дисків. Віртуальні диски виступають як окремий жорсткий диск для віртуальних машин, що перебувають на жорсткому диску хост-системи. Скориставшись інтерфейсом (рис. 3.19) створення нового віртуального диску виділимо для нього 10 гігабайт.

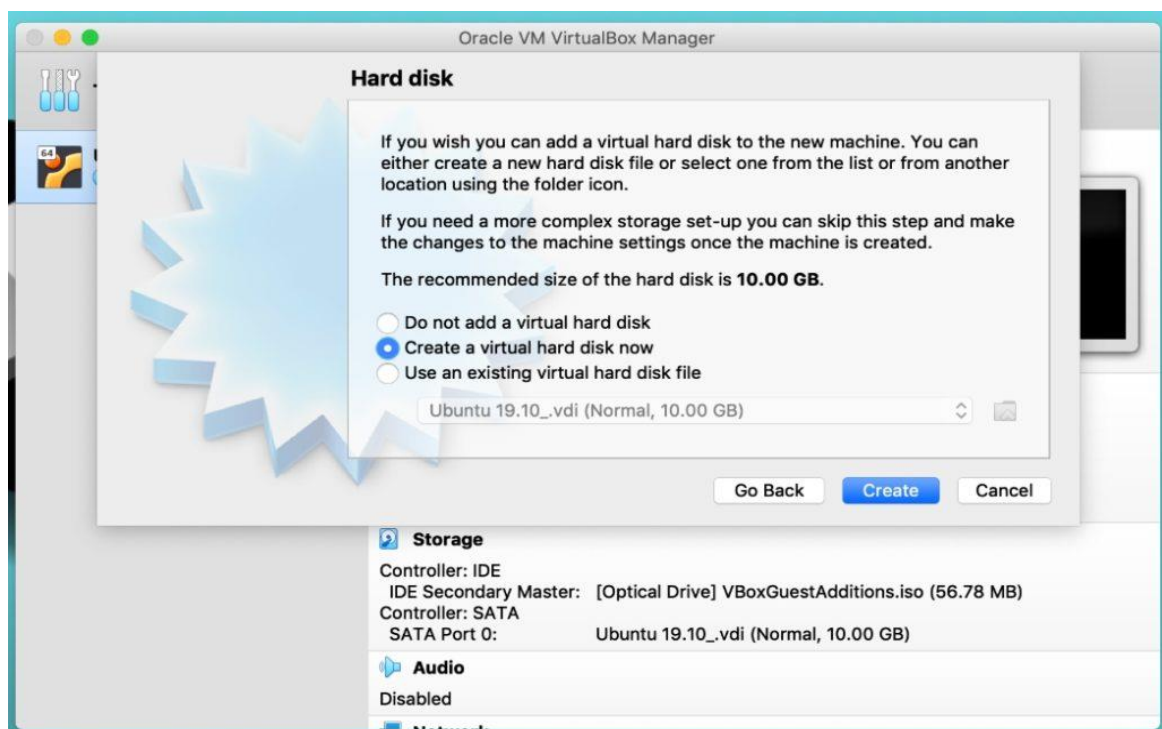


Рисунок 3.19 – Вікно створення нового віртуального диску

5. Після створення віртуального диска в наступному вікні (рис. 3.20) буде запропоновано вибрати тип файлу. Він вирішує спосіб зберігання віртуальної машини у файлі. Вибираємо VDI, який є власним форматом файлу VirtualBox. Він створює єдині файли .VDI і зберігає цілу віртуальну операційну систему.



Рисунок 3.20 – Вікно вибору типу файлу нового віртуального диску

6. Далі потрібно визначити масштабованість віртуального диска (рис. 3.21). У майбутньому, якщо буде потрібно, то можна збільшити простір на диску, динамічне розподіл може дозволити це зробити. Але якщо буде вибраний фіксований розмір, то після створення віртуальної машини не буде можливо додати додатковий простір.

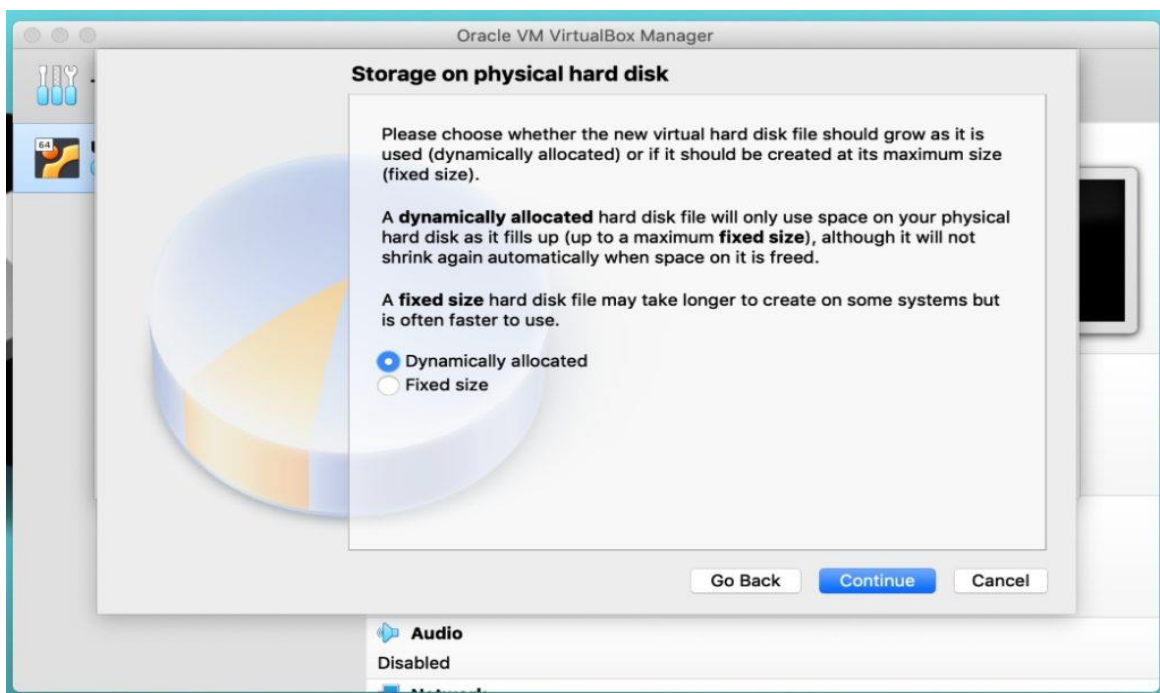


Рисунок 3.21 – Вікно вибору типу масштабованості

7. Оскільки вибравши тип файлу VDI, потрібно зберігати цей єдиний файл у хост-системі. Потрібно зазначити місце (рис. 3.22) для зберігання і виділити місце відповідно до вимог. Ubuntu рекомендує щонайменше 25 ГБ сховища для безперебійної роботи, отже, встановимо 30 ГБ. У світі віртуальних машин операційна система, що фактично працює на нашому комп'ютері, називається хостом, а будь-яка операційна система, що працює в віртуальних машинах, - гостями

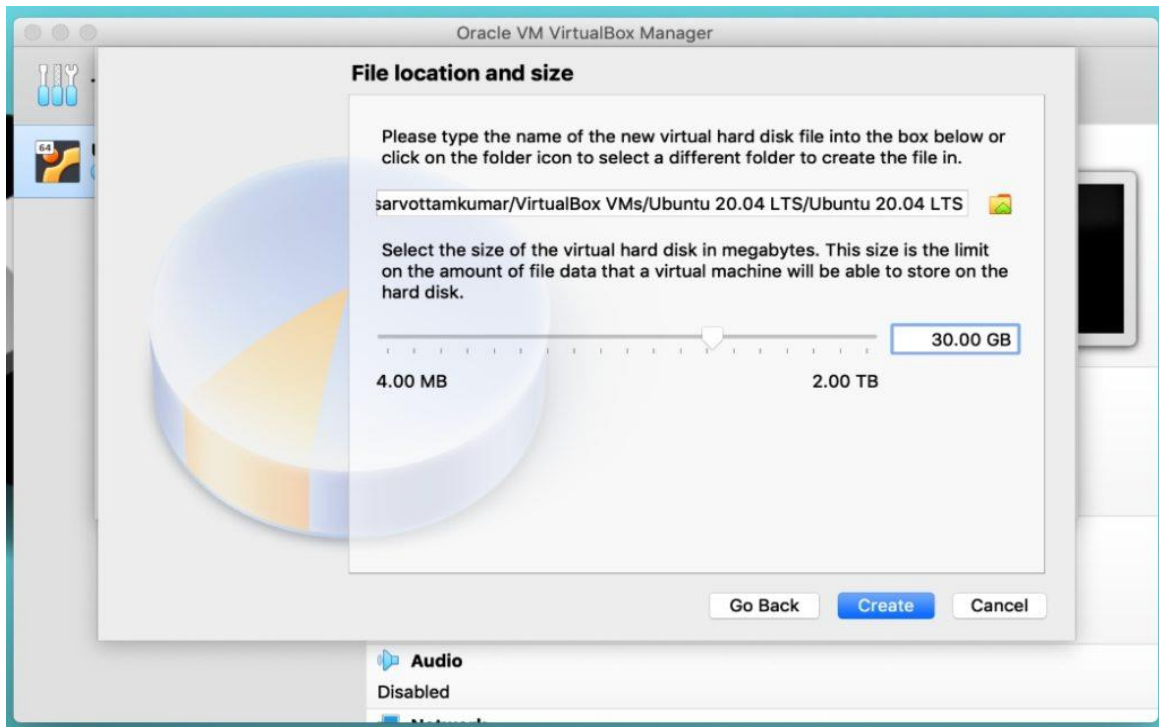


Рисунок 3.22 – Вікно вибору розміру файла

8. Після натискання на «Create» наша машина створена (рис. 3.23).



Рисунок 3.23 – Вікно головного інтерфейсу VirtualBox



9. Завантажимо Ubuntu 20.04 ISO файл у Virtual Machine. Перейдемо до розділу «Storage» (рис. 3.24), а потім натиснемо на піктограму диска, щоб імпортувати завантажений образ Ubuntu 20.04.

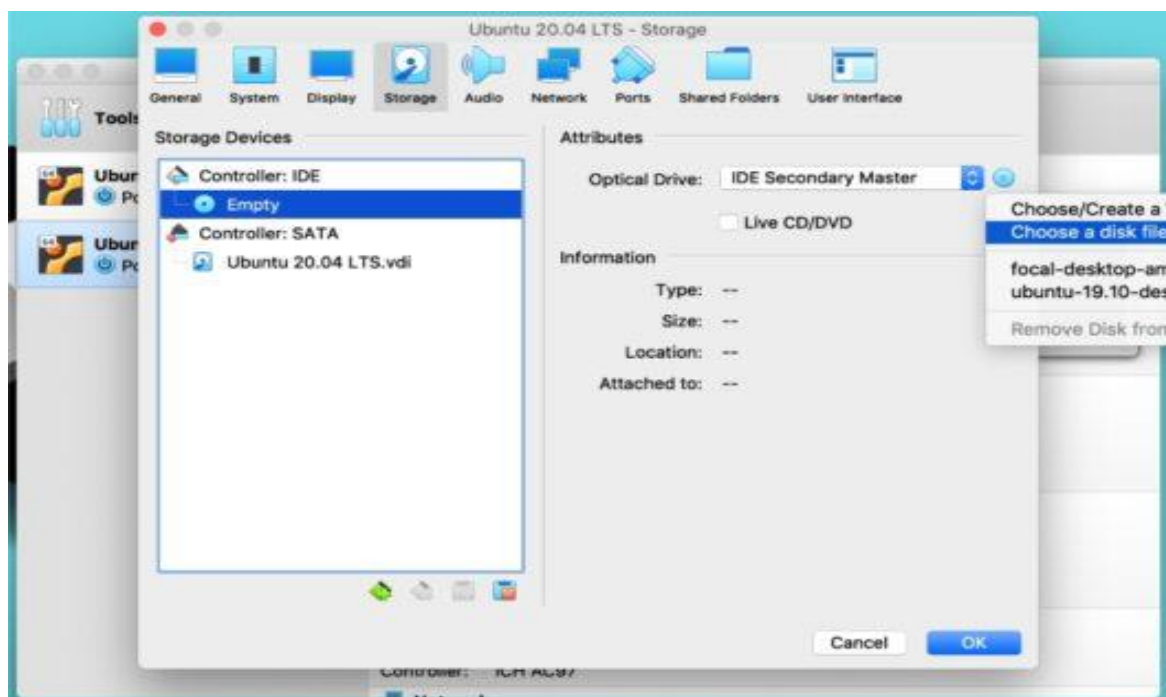


Рисунок 3.24 – Вікно інтерфейсу Storage

Після натиснення на «Choose a disk file», відкриється діалогове вікно. Вибравши Ubuntu ISO з діалогового вікна та завантаживши його у віртуальну машину. Потрібно натиснути «ОК», щоб зберегти все. Далі потрібно розпочати встановлення Ubuntu.

10. Для встановлення Ubuntu 20.04 користувач має вибрати віртуальну машину зі списку (рис. 3.25) та вибрати раніше завантажений образ дику з операційною системою. Натиснувши «Start» починає працювати Ubuntu installer (рис. 3.26). Замість фізичної машини, за допомогою апаратної віртуалізації, операційна система розподіляє ресурси головного комп'ютера. Кожна ВМ матиме власну операційну систему (ОС), яка працює окремо від головної ОС. Так само кожна віртуальна машина матиме власні процесори, оперативну пам'ять, диск тощо. Завдяки віртуалізації це, по суті, дві різні машини.



Рисунок 3.25 – Вікно вибору образу ОС для встановлення

У конкретній віртуальній машині гостьова ОС зберігається на віртуальному жорсткому диску - великому файлі з кількома гігабайтами, що знаходиться на нашому реальному жорсткому диску.



Рисунок 3.26 – Вікно інтерфейсу встановлення Ubuntu



11. Далі вибравши «Install Ubuntu» (рис. 3.27) потрібно виконати базові інструкції як і при локальному встановленні системи (рис. 3.28).



Рисунок 3.27 – Вікно вибору режиму встановлення Ubuntu

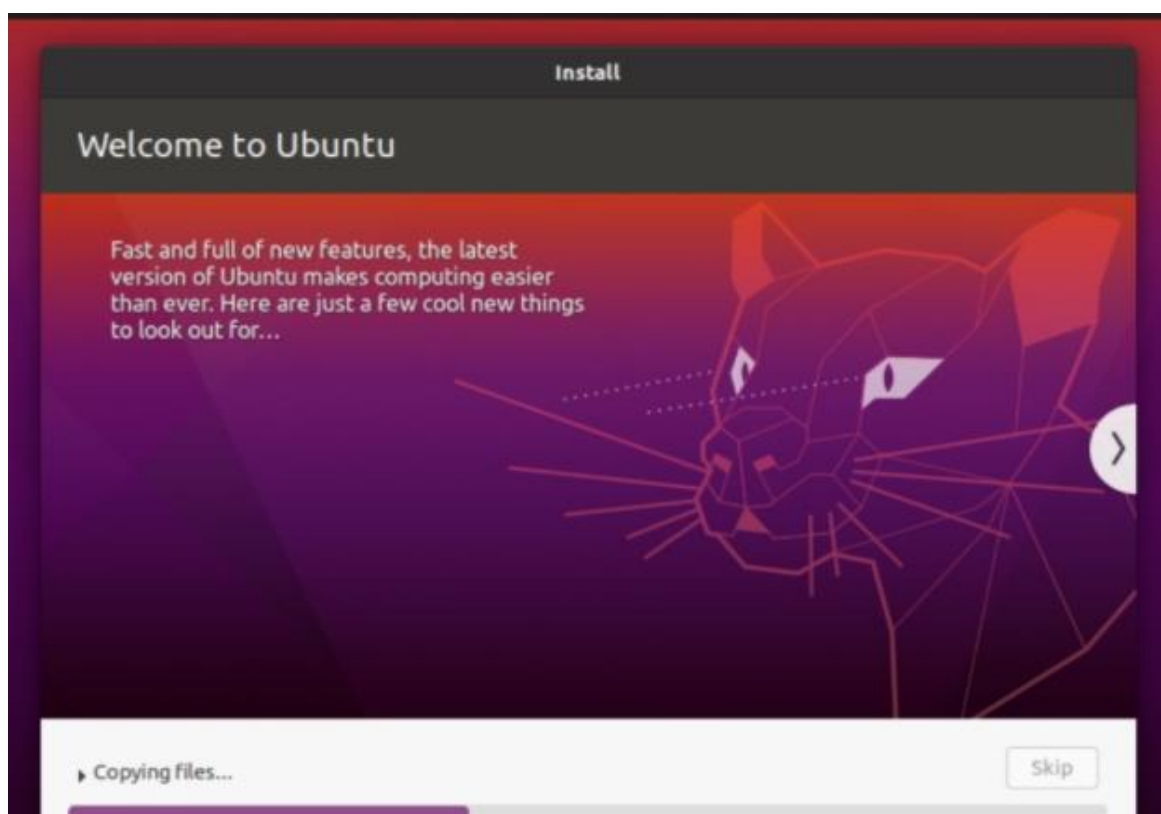


Рисунок 3.28 – Вікно стартового інтерфейсу Ubuntu

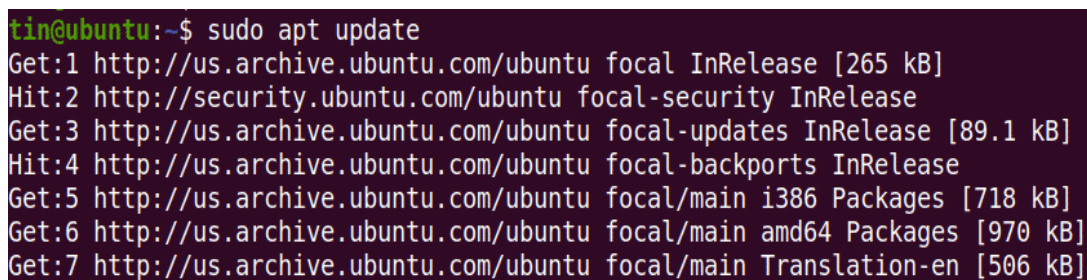
### 3.3.3 Встановлення локального web-сервера Apache2

HTTP-сервер Apache є безкоштовним програмним забезпеченням для веб-серверів із платформою з відкритим кодом, випущеним на умовах Apache License 2.0. Apache розробляється та підтримується відкритою спільнотою розробників під егідою Фонду програмного забезпечення Apache.

Переважна більшість екземплярів сервера Apache HTTP працює на дистрибутиві Linux.

Для того, щоб встановити Apache2, ми повинні виконати наступні дії:

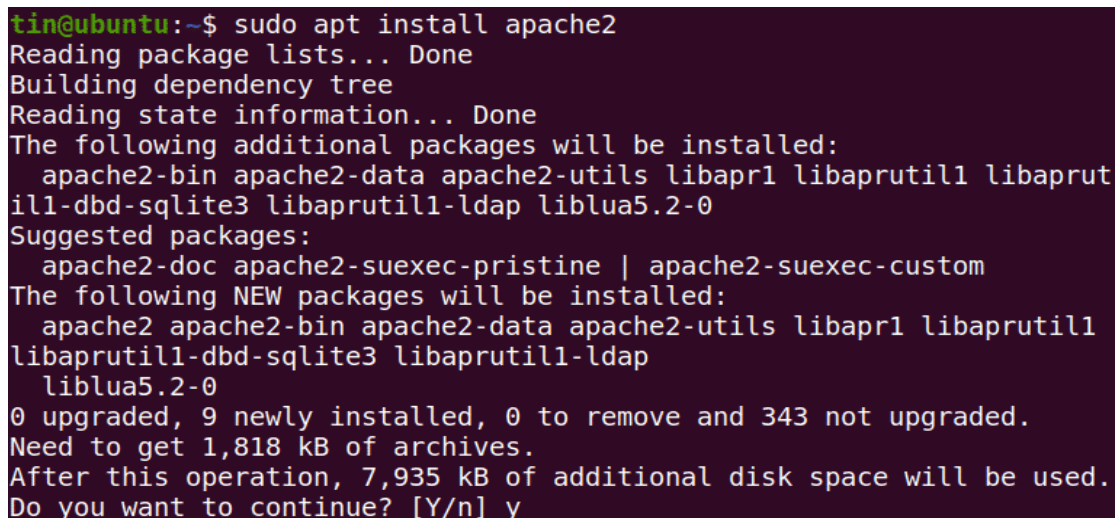
1. По-перше, потрібно оновити індекс системного сховища, щоб встановити найновішу версію Apache2. Для цього має бути виконана наступна команда: «sudo apt update» (рис. 3.29).



```
tin@ubuntu:~$ sudo apt update
Get:1 http://us.archive.ubuntu.com/ubuntu focal InRelease [265 kB]
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease [89.1 kB]
Hit:4 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease
Get:5 http://us.archive.ubuntu.com/ubuntu focal/main i386 Packages [718 kB]
Get:6 http://us.archive.ubuntu.com/ubuntu focal/main amd64 Packages [970 kB]
Get:7 http://us.archive.ubuntu.com/ubuntu focal/main Translation-en [506 kB]
```

Рисунок 3.29 – Вікно оновлення індексів системного сховища

2. Безпосереднє встановлення Apache2. Для цього виконується команда терміналу: «sudo apt install apache2» (рис. 3.30)

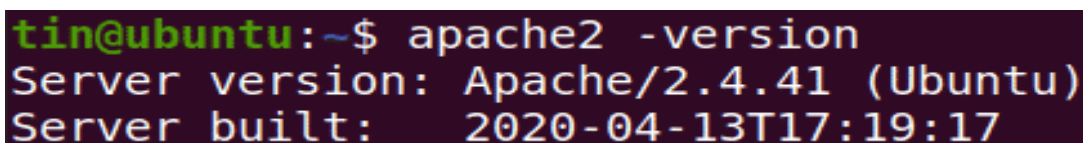


```
tin@ubuntu:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
0 upgraded, 9 newly installed, 0 to remove and 343 not upgraded.
Need to get 1,818 kB of archives.
After this operation, 7,935 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Рисунок 3.30 – Вікно встановлення Apache2

Система може попросити підтвердження, надавши варіант «Y/n.» Потрібно вибрати «у», а потім натиснути Enter, щоб продовжити. Після цього веб-сервер Apache2 та всі його залежності будуть встановлені.

3. Потрібно перевірити, що веб-сервер Apache2 встановлений вірно. Для цього виконується команда в терміналі: «apache2 -version» (рис. 3.31)



```
tin@ubuntu:~$ apache2 -version
Server version: Apache/2.4.41 (Ubuntu)
Server built: 2020-04-13T17:19:17
```

Рисунок 3.31 – Вікно встановлення Apache2

4. Налаштування веб-сервера Apache. Перш ніж рухатися до налаштувань, спочатку потрібно перевірити, чи працює Apache сервіс. Для цього виконаємо наступну команду в терміналі: «sudo systemctl status apache2».

5. Потрібно створити директорію для домена. Для цього виконується команда: «sudo mkdir -p /var/www/fa-fa»

6. Зміна власника каталогу на поточного користувача виконується наступною командою: «sudo chown -R \$USER:\$USER /var/www/fa-fa»

7. Встановлення необхідних права відбувається наступним чином: «sudo chmod -R 755 /var/www/fa-fa»

8. Створення virtual host file. Для цього виконується наступна команда:

«sudo nano /etc/apache2/sites-available/fa-fa.net.conf» та введемо наступні рядки:

```
<VirtualHost *:80>
ServerAdmin admin@fa-fa.net
ServerName fa-fa.net
ServerAlias fa-fa.net
DocumentRoot /var/www/fa-fa.net/html
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

9. Для активації virtual [18] host configuration file виконається команда в терміналі: «sudo a2ensite fa-fa.net.conf». Також потрібна деактивація конфігурації за

замовчуванням: «sudo a2dissite 000-default.conf». А після потрібно перезавантажити Apache для активації нової конфігурації: «sudo systemctl restart apache2»

10. Для тестування потрібно ввести адресу веб-додатку (рис. 3.32),



Рисунок 3.32 – Вікно браузера з відображенням fa-fa.net

Налаштування веб-сервер Apache2 завершено (рис 3.33).

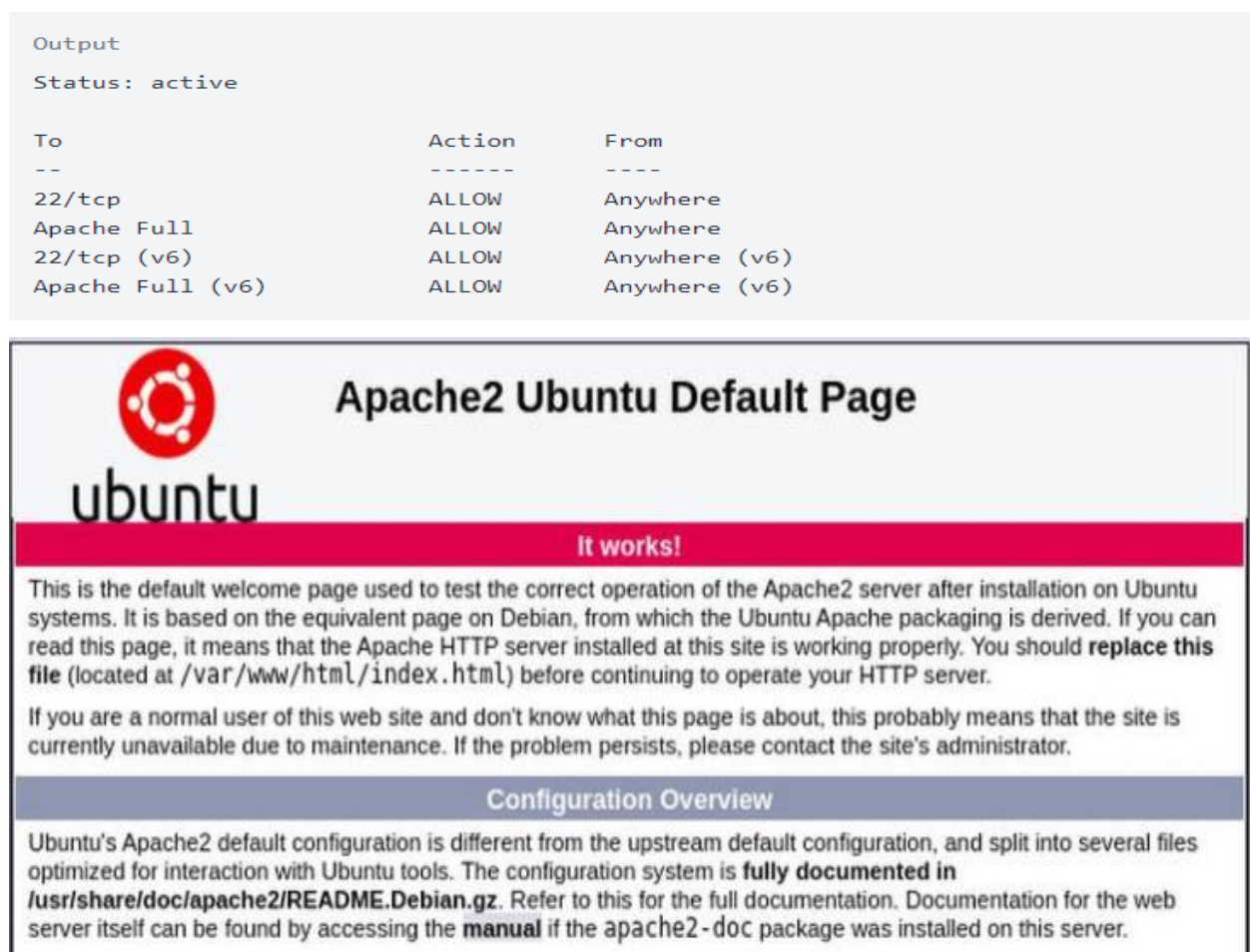


Рисунок 3.33 – Вікно Apache2

### 3.3.4 Встановлення Framework Laravel

Laravel в основному використовується для створення власних веб-програм за допомогою PHP. Це веб-фреймворк, який вирішує багато проблем, які раніше потрібно було робити вручну, наприклад, маршрутизацію, шаблонування HTML та автентифікацію.

Laravel повністю серверний фреймворк, завдяки запуску на PHP, і зосереджений на обробці даних та дотриманні дизайну Model-View-Controller.

Такий фреймворк, як React, може приділяти більшу частину своєї уваги до взаємодії з користувачами в той час як Laravel створює міцну основу для того, щоб розробник міг її нарощувати - і робити продукт легко масштабованим.

Для початку потрібно завантажити Composer в папку проекту. Для цього необхідно створити порожню папку в root-папці веб-сервера Apache і назвати її `fafa_project` для цього використаємо наступні команди:

```
cd /var/www  
mkdir fafa_project  
cd fafa_project
```

Завантажимо Composer в цю папку за допомогою команди:

```
curl -sS https://getcomposer.org/installer | php
```

На екрані має відобразитись повідомлення про успішне виконання завдання:

```
All settings correct for using Composer
```

```
Downloading...
```

```
Composer successfully installed to: /var/www/fafa_project/composer.phar
```

```
Use it: php composer.phar
```

Тепер, перейшовши в папку проекту, можна побачити всього один файл, `composer.phar`; це архівний файл PHP, який можна запустити за допомогою терміналу.

Щоб завантажити Composer (рис. 3.34) в папку проекту, не виходячи при цьому з поточного каталогу, потрібно вказати опцію `-install-dir`.



```

Changed current directory to /home/ubuntu/.config/composer
Using version ^3.1 for laravel/installer
./composer.json has been created
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 20 installs, 0 updates, 0 removals
- Installing symfony/polyfill-php80 (v1.17.0): Downloading (100%)
- Installing symfony/process (v5.1.0): Downloading (100%)
- Installing symfony/polyfill-ctype (v1.17.0): Downloading (100%)
- Installing symfony/filesystem (v5.1.0): Downloading (100%)
- Installing symfony/polyfill-mbstring (v1.17.0): Downloading (100%)
- Installing symfony/polyfill-intl-normalizer (v1.17.0): Downloading (100%)
- Installing symfony/polyfill-intl-grapheme (v1.17.0): Downloading (100%)
- Installing symfony/string (v5.1.0): Downloading (100%)
- Installing psr/container (1.0.0): Downloading (100%)
- Installing symfony/service-contracts (v2.1.2): Downloading (100%)
- Installing symfony/polyfill-php73 (v1.17.0): Downloading (100%)
- Installing symfony/console (v5.1.0): Downloading (100%)
- Installing ralouphie/getallheaders (3.0.3): Downloading (100%)
- Installing psr/http-message (1.0.1): Downloading (100%)
- Installing guzzlehttp/psr7 (1.6.1): Downloading (100%)
- Installing guzzlehttp/promises (v1.3.1): Downloading (100%)
- Installing symfony/polyfill-php72 (v1.17.0): Downloading (100%)
- Installing symfony/polyfill-intl-idn (v1.17.0): Downloading (100%)
- Installing guzzlehttp/guzzle (6.5.4): Downloading (100%)
- Installing laravel/installer (v3.1.0): Downloading (100%)
symfony/polyfill-intl-normalizer suggests installing ext-intl (For best performance)
symfony/polyfill-intl-grapheme suggests installing ext-intl (For best performance)
symfony/service-contracts suggests installing symfony/service-implementation
symfony/console suggests installing symfony/event-dispatcher
symfony/console suggests installing symfony/lock
symfony/console suggests installing psr/log (For using the console logger)
guzzlehttp/psr7 suggests installing zendframework/zend-httphandler (Emit PSR-7 responses)
symfony/polyfill-intl-idn suggests installing ext-intl (For best performance)
guzzlehttp/guzzle suggests installing psr/log (Required for using the Log middleware)
Writing lock file
Generating autoload files
13 packages you are using are looking for funding.
Use the `composer fund` command to find out more!

```

Рисунок 3.34 – Встановлення залежностей через Composer

```

curl -sS https://getcomposer.org/installer | php -- --install-
dir=/var/www/fafa_project

```

Ця команда дозволяє завантажувати файли в зазначений каталог (в даному випадку це fafa\_project) з будь-якої точки системи. Щоб переконатися, що Composer працює належним чином, використовуйте наступну команду, яка виведе список доступних команд: `php composer.phar`

Використовуючи Composer завантажимо інсталятор Laravel: `composer global require "laravel/installer=~1.1"`

Встановимо Laravel Framework за допомогою команди: `composer create-project laravel/laravel --prefer-dist`.

Після виконання вище наведених команд фреймворк Laravel встановлено і можна розпочати розробку та проектування системи.

### 3.4 Проектування та налаштування інфраструктури AWS

Amazon Web Services (AWS), хмарна платформа, пропонована Amazon.com Inc (AMZN), стала гігантською складовою бізнес-портфеля гіганта електронної комерції. У першому кварталі 2020 року AWS принесла рекордну виручку в розмірі 10 мільярдів доларів, що склало 13,5% загального доходу Amazon. Протягом останніх кількох кварталів AWS стабільно зростає у 30-відсотковому діапазоні, і це передовик інших платформ хмарних обчислень, таких як конкурент Microsoft Azure.

В загальному розумінні AWS представляє собою центральні процесорні блоки (ЦП) та графічні процесори (ГП) для обробки; локальну та оперативну пам'ять; сховище на жорсткому диску SSD; вибір операційних систем; створення мереж; та попередньо завантажене прикладне програмне забезпечення, таке як веб-сервери, бази даних та управління взаємовідносинами з клієнтами (CRM).

AWS пропонує кілька способів швидко розпочати роботу за допомогою загальних сценаріїв розгортання; справді, часто існує більше ніж один спосіб реалізації конкретного рішення. Щоб краще зрозуміти хмарний-сервіс AWS, має сенс розглянути список найбільш часто використовуваних компонентів.

EC2 віртуальні сервери (Elastic Compute Cloud) складають основу хмарного розгортання на AWS. Ці віртуальні сервери доступні у різних конфігураціях, кожна з різною кількістю процесорів, пам'яті, пропускною здатністю мережі, та тарифікується на основі погодинної ставки.

S3 (Simple Storage Service) - це система зберігання об'єктів, яка може зберігати до 5 ТБ в одному об'єкті і доступна в будь-якій точці Інтернету через необхідні оператори командного рядка, виклики API або за допомогою програм, призначені для роботи з ним.

EBS (Elastic Block Store) пропонує традиційні можливості файлової системи і є дорожчим. Приєднані до сервера, томи EBS функціонують як накопичувач і зберігаються навіть після вимкнення обчислювального екземпляра.

RDS (Relationship Database Service) - це веб-служба, яка спрощує налаштування, експлуатацію та масштабування реляційної системи управління базами даних

(СУБД). Можна вибрати відповідний провайдер баз даних, включаючи MySQL, Oracle, SQL Server, PostgreSQL або Amazon Aurora.

Route 53 для реєстрації DNS та імен доменів. Route 53 пропонує конкурентоспроможні ціни, які можуть бути дешевшими, ніж ціни, пропоновані деякими реєстраторами доменних імен.

Отже розпечемо з вибору типу і розміру образу та вибору регіону.

AWS дозволяє легко запустити свій перший обчислювальний екземпляр, пропонуючи широкий спектр попередньо побудованих та оптимізованих образів Amazon Machine Images (AMI), які ми можете завантажити на щойно створений екземпляр. AWS Marketplace також існує для сторонніх створених зображень, а спільнота AWS також створює та використовує спільні загальнодоступні AMI.

Перш ніж налаштовувати нашу хмарну інфраструктуру, нам потрібно буде вибрати місце або “регіон”, з якого буде базуватися наша віртуальна інфраструктура. Ідея тут полягає в тому, щоб вибрати місце, яке є або найближчим до основної маси ваших користувачів, або найближчим до місця, де фізично розташовані наші розробники. Для останнього це може призвести до дещо кращого та швидшого завантаження даних та розробки веб-сайту. Розробники та адміністратори баз даних мають знати, що синхронна реплікація бази даних не підтримується в різних регіонах.

Звичайно, залежно від архітектури нашого веб-сайту, використання хорошої служби Content Delivery Network (CDN) може в більшості випадків зробити наш регіон розгортання спірним. Використаємо AWS CloudFront CDN, хоча можна використовувати й інші варіанти. Крім того, AWS пропонує інструменти для легкої міграції між різними регіонами.

Варто зазначити, що, хоча вартість більшості послуг AWS зазвичай однакова в різних регіонах, це не завжди так. В нашому випадку ми виберемо європейський регіон «eu-central-1».

Загалом, ми повинні забезпечити, щоб важливі компоненти системи могли працювати з більш ніж однієї зони доступності (AZ) в межах регіону. Як правило, це передбачає встановлення серверної бази даних для розгортання декількох AZ з самого



початку. Подібно до того, як мати більше ніж один сервер баз даних у локальній інфраструктурі.

Найбільш типове налаштування передбачає налаштування Elastic Load Balancer (ELB) для розподілу вхідного трафіку додатків між кількома екземплярами обчислень. Трафік може бути автоматично перенаправлений з «нездорових» екземплярів на «здорові», які можуть охоплювати кілька AZ у разі аварії.

AWS серйозно ставиться до безпеки, і це не дивно, оскільки є можливість налаштувати буквально сотні виробничих серверів - або видалити їх – одним натиском миші.

Для кращого управління безпекою AWS рекомендує налаштовувати користувачів з обмеженими дозволами для управління ресурсами, за які вони платять, на відміну від root користувача з необмеженим доступом. Як і в типовій системі Linux, користувачів можна розподіляти по групах, тоді як додаткові ролі можна створювати та призначати користувачам або групам.

Крім того, AWS також пропонує багатофакторну автентифікацію (MFA), яка доступна як в апаратних, так і у віртуальних варіантах. Для апаратного M3C AWS підтримує захисні вставки, виготовлені Gemalto, стороннім постачальником. Крім того, підтримується віртуальна програма MFA, Google Authenticator підтримується як опція на Android, iPhone та BlackBerry, а програма AWS Virtual MFA на Android.

Важливим аспектом хмарних обчислень є його здатність зменшити вартість інфраструктури. Однак, збільшення операційних витрат за певних обставин може перевищити вартість локального розгортання у відносно короткій перспективі.

Щоб допомогти користувачам отримати більш точне уявлення про вартість їх хмарних сервісів, AWS розробив щомісячний калькулятор, за допомогою якого користувачі можуть обчислювати вартість своїх систем на основі служб, якими вони користуються, відповідно до їх передбачуваного рівня використання диска та мережі. Це може допомогти компаніям вирішити, чи можна їм обійтися без певного рівня надійності та додаткових послуг.

Для того щоб оптимізувати свої витрати на основі існуючого розгортання та реалізувати структурну схему з Додатку А, потрібно буде використати spot екземпляри, що дозволить попередньо бронювати та сплачувати за них заздалегідь.

AWS пропонує послугу Trusted Advisor, яка допоможе налаштувати різні аспекти розгортання AWS, включаючи безпеку та оптимізацію витрат. Однак для розблокування всіх рекомендацій потрібен платний план підтримки.

Безпека на AWS починається зі створення власної Amazon Virtual Private Cloud (VPC) - виділеної віртуальної мережі, яка розміщує ваші ресурси AWS і логічно ізольована від інших віртуальних мереж у хмарі AWS. VPC отримує власний діапазон IP-адрес, повністю налаштування підмережі, таблиці маршрутизації, списки контролю доступу до мережі та групи безпеки (фактично брандмауер). Для того щоб захистити веб-сайт, потрібно створивши правила груп безпеки, які дозволяють веб-серверу реагувати на вхідні HTTP та SSL-запити з Інтернету, одночасно забороняючи веб-серверу ініціювати вихідні підключення до Інтернету (рис. 3.35).

SSL був створений для виправлення проблеми захисту конфіденційності користувачів. Шифруючи будь-які дані, які переходять між користувачем та веб-сервером, SSL гарантує, що кожен, хто перехоплює дані.

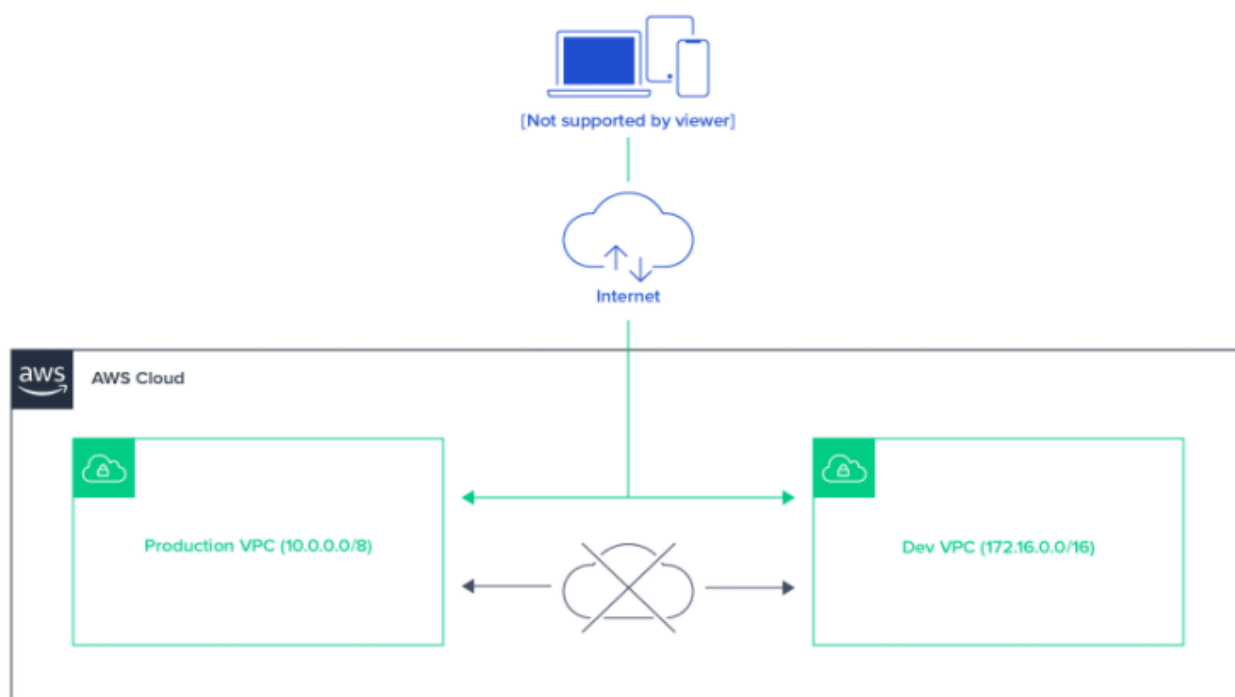


Рисунок 3.35 – Базова схема AWS VPC

Налаштування двох VPC наведених в Додатку Г ізолює виробниче середовище від решти середовищ на рівні мережі, виключаючи випадкову неправильну конфігурацію програми, яка не може перейти цю межу. Навіть якщо конфігурація виробничого середовища помилково вказує на ресурси середовища розробки, отримати доступ до них неможливо.

Завдяки тому що середовища розробки, тестування та контролю якості, мають один і той же VPC, можливий транскордонний доступ у разі неправильної конфігурації, але, оскільки ці середовища використовують тестові дані, реальних проблем з безпекою не існує.

Враховуючи вище наведену інформацію та ресурсні потреби системи було розроблено архітектурну схему розгортання системи на AWS наведену в Додатку Ж. Використовуючи Amazon VPC для аварійного відновлення, ми отримуємо всі переваги відновлення системи в аварійній ситуації за незначні кошти.

### 3.5 Проектування бази даних

Проектування бази даних – це сукупність процесів, що виконують наступні задачі: проектування, розробку, впровадження та обслуговуванню систем управління даними. Правильно розроблена база даних проста в обслуговуванні, покращує узгодженість даних та економічно вигідна з точки зору місця використання ресурсів. Дизайнер баз даних вирішує, як елементи даних співвідносяться та які дані повинні зберігатися.

Основними цілями проектування [21] баз даних є створення логічних та фізичних моделей конструкцій запропонованої системи баз даних.

Логічна модель концентрується на вимогах до даних та даних, що зберігаються незалежно від фізичних міркувань. Це не стосується того, як дані будуть зберігатися або де вони будуть зберігатися фізично.

Модель фізичного проектування даних (рис. 3.36) передбачає переклад логічного дизайну бази даних на фізичний носій за допомогою апаратних ресурсів та програмних систем, таких як системи управління базами даних (СУБД). Для

правильного проектування фізичної бази даних важливо розуміти, як і як часто буде здійснюватися доступ до даних. В ідеалі моделі процесів повинні містити посилання на бізнес-функції, які вказують, як часто повинен здійснюватися бізнес-процес. Це можна перевести на псевдокод.

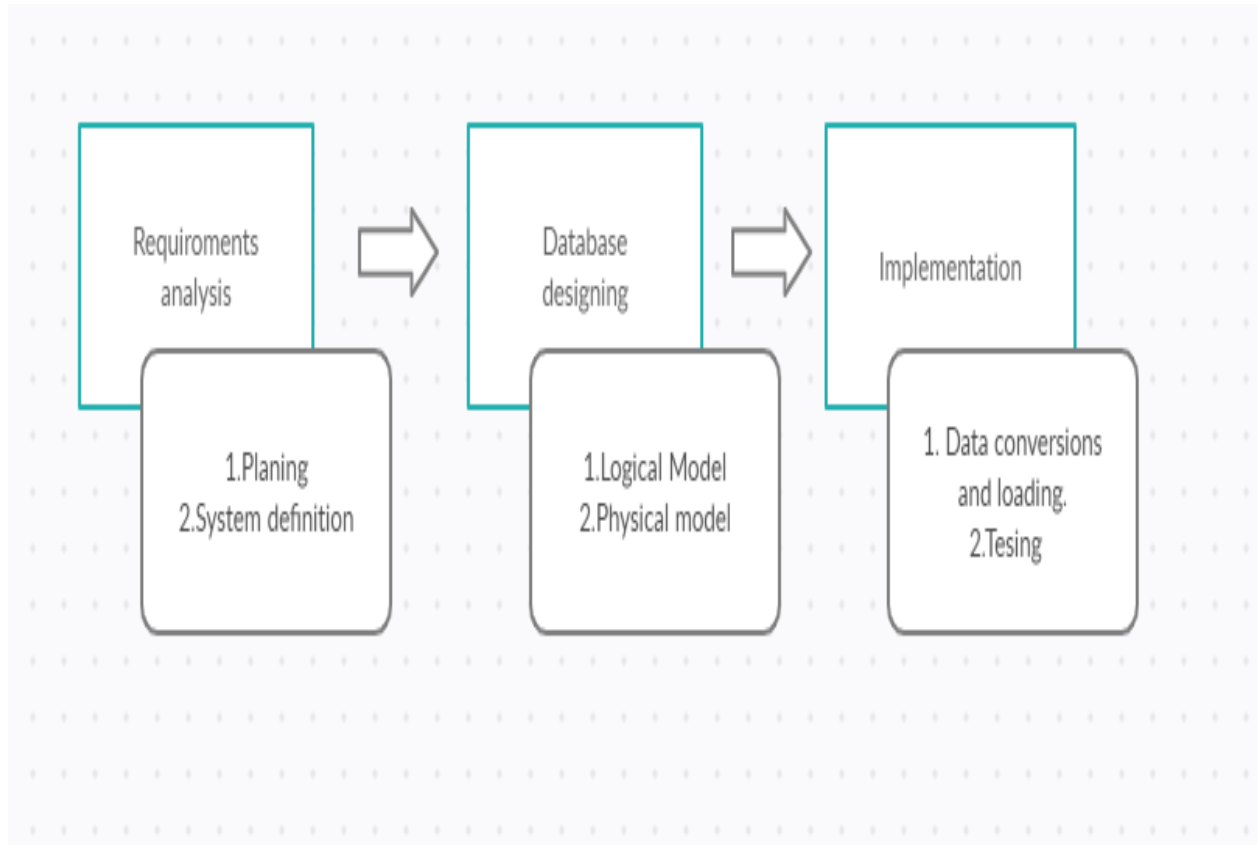


Рисунок 3.36 – Модель проектування даних

Одним із найважливіших кроків в проектуванні баз даних є нормалізація. Нормалізація – це техніка проектування бази даних, яка зменшує надмірність даних та усуває такі небажані характеристики, як аномалії вставки, оновлення та видалення. Правила нормалізації поділяють більші таблиці на менші таблиці та пов'язують їх за допомогою зв'язків. Метою нормалізації в SQL є усунення зайвих (повторюваних) даних та забезпечення логічного збереження даних.

Винахідник реляційної моделі Едгар Кодд запропонував теорію нормалізації з введенням Першої нормальної форми, і він продовжував поширювати теорію на Другу і Третю нормальну форми. Пізніше він приєднався до Реймонда Ф. Бойса для розробки теорії Бойса-Кодда Нормала [10]

Перелік нормальних форм БД:

- 1NF (перша нормальна форма);
- 2NF (друга нормальна форма);
- 3NF (третя нормальна форма);
- BCNF (нормальна форма Бойса-Кодда);
- 4NF (четверта нормальна форма);
- 5NF (п'ята звичайна форма);
- 6NF (шоста нормальна форма).

Теорія нормалізації даних у SQL все ще розвивається. Наприклад, існують дискусії навіть у 6-й звичайній формі. Однак у більшості практичних застосувань нормалізація досягає найкращих результатів у 3-й нормальній формі.

Проектування баз даних має вирішальне значення для успішного впровадження системи управління базами даних, яка відповідає вимогам до даних корпоративної системи. Нормалізація в СУБД допомагає створювати системи баз даних, які є економічно вигідними та мають кращі моделі безпеки. Функціональні залежності є дуже важливою складовою процесу нормалізації даних. Більшість систем баз даних є нормалізованими базами даних до третьої нормальної форми.

Первинний ключ однозначно ідентифікує запис у таблиці і не може бути нульовим. Зовнішній ключ допомагає зв'язати таблицю та посиляється на первинний ключ

Багато разів ми зустрічаємо дані з обмеженою ієрархією. Наприклад, розглянемо багаторівневий маркетинговий сценарій, коли продавець може мати кількох продавців нижче себе. Для таких сценаріїв використання первинного та зовнішнього ключа, що посиляється на себе, допоможе досягти бажаної мети.

Враховуючи кількість унікальних коефіцієнтів розроблюваної системи була спроектована наступна модель бази даних наведено у Додатку Б.

В цій моделі враховано збереження всіх коефіцієнтів які впливають на обрахунок кінцевої вартості страхового поліса:

- K1 (Vehicle type) – зі зрозумілих причин такі висококласні автомобілі, як Audi і Bentley, застраховані за вищою ціною, ніж бюджетні машини, такі як Lada та Kia. На

подібних підставах позашляховики застраховані на вищій премії, ніж сімейний автомобіль;

- K2 (Region) – регіон реєстрації автотранспорту, оскільки щільність руху в міських районах вища, автомобілі застраховані за дещо вищою премією в міських районах;
- K3 (Owner Type) – тип власника автотранспорту (фізична чи юридична особа);
- K4 (Experience) – особи, які не досягли 25 років, частіше потрапляють у нещасні випадки, а отже, особи віком від 18 до 25 років повинні сплачувати вищу премію;
- K5 (Policy Period) – термін на який оформлюється страховий поліс, чим довше тим вигідніше для користувача;
- K6 (Taxi) – використання автотранспорту в таксі збільшує вартість страхового поліса;
- K7 (Under 1 year) – додатковий коефіцієнт для врахування водійського стажу до 1-го року;
- K8 (Bonus-Malus) – коефіцієнт для додаткового зменшення або збільшення вартості страхового поліса враховуючи частоту звернень користувача;
- K9 (Discount) – коефіцієнт ручного встановлення знижки для користувача. Дозволяє адміністратору системи встановлювати персональні знижки.

Завдяки нормалізації даних для всіх, хто отримує доступ до БД значно збільшується надійність, а також дотримується більша послідовність інформації. Це дозволяє уникнути таких помилок як зберігання застарілих версій, дублювання донних, що знаходяться в різних таблицях системи, та різні типи зав'язків між даними про продукт, що виникають без чіткої ієрархії.

Зберігання та відображення даних має більш логічну організацію, і тому корисність для будь-якого підрозділу системи, що використовує таблиці, значно зростає. Нормалізація даних означає, що інформація завжди організовується і зберігається у своєму належному місці, без дублікатів або застарілих версій.

Набагато простіше підтримувати існуючі бази даних та їх розширювати. Також набагато швидше підключити джерела даних до будь-якої внутрішньої чи зовнішньої

системи, оскільки це не потребує перевірки того, що данні мають правильну структуру.

Порівняно з системою, в якій не приділяється належна увага до бази даних, з системою, де було проведено нормалізацію даних, можна запропонувати клієнтам додаткові гарантії цілісності даних.

Основні переваги нормалізації в аналітичному плані полягають у тому, що вона дозволяє швидше шукати та сортувати, та краще створювати індекси для менших логічних таблиць. Також, маючи більше таблиць, можна краще використовувати сегменти для контролю фізичного розміщення даних.

### 3.6 Розробка Frontend

Для розробки інтерактивного графічного інтерфейсу використано React та HTML5.

React - найпопулярніша інтерфейсна бібліотека JavaScript у галузі веб-розробки. Її використовують як великі продуктові компанії так і соціальні мережі (Netflix, Airbnb, Instagram та New York Times,).

React приносить таблиці багато переваг, що робить його кращим вибором, ніж інші фреймворки, такі як Angular.js..

Однак React займається лише наданням даних у DOM, тому створення React-програм зазвичай вимагає використання додаткових бібліотек для управління станом та маршрутизації. Redux та React Router [20] є відповідними прикладами таких бібліотек.

React дозволяє писати модульний і чистий код, розбиваючи проект на окремі компоненти.

Завдяки модульній структурі код реагування простий в обслуговуванні та масштабуванні, що заощаджує час розробників у довгостроковій перспективі.

Віртуальний dom та рендеринг на стороні сервера дозволяють розробникам створювати масштабовані додатки.

ReactJS це бібліотека з відкритим кодом, тому технологія має багату екосистему з безліччю пакетів для використання.

Для швидкої та легкої роботи з DOM (рис. 3.37) розробники React придумали побудувати віртуальний DOM.

Віртуальний DOM дозволяє React знати, коли повторно відтворювати або ігнорувати конкретні фрагменти DOM, оскільки він може відстежувати, коли ці дані змінюються.

Для розробки графічного інтерфейсу системи нам стане у нагоді діаграма компонентів наведена у Додатку Е.

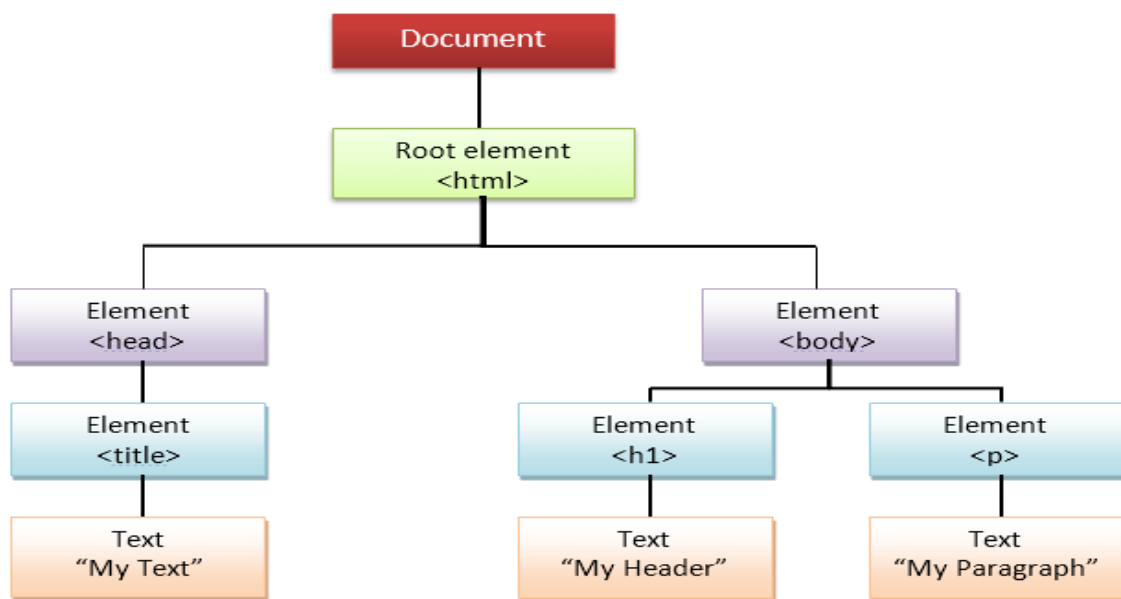


Рисунок 3.37 – DOM дерево

Що робить React настільки потужним – це ідея компонентів. Розробникам не потрібно турбуватися про цілий веб-додаток. Його можна розбити на прості компоненти і в подальшому їх використовувати повторно.

Компонент – це простий клас Js, який успадковується від `React.Component` і реалізує метод візуалізації. Цей метод візуалізації просто повинен повернути JSX для HTML, який в подальшому він повинен відтворити.

JSX дозволяє нам писати елементи HTML у JavaScript та розміщувати їх у DOM без будь-яких методів `createElement()` та / або `appendChild()`.

Оскільки JSX ближче до JavaScript, ніж до HTML, React DOM використовує конвенцію про іменування властивостей `camelCase` замість імен атрибутів HTML.



Наприклад, `class` стає `className` у JSX, а `tabindex` стає `tabIndex`. JSX запобігає ін'єкційним атакам, тому це безпечно будувати елементи вводу даних від користувача в JSX.

За замовчуванням React DOM уникає будь-яких значень, вбудованих у JSX, перед їх відтворенням. Таким чином, це гарантує, що ви ніколи не зможете використати JS ін'єкцію, що чітко не прописано у нашому додатку.

Що стосовно компонентів (Components), то вони дозволяють розділити інтерфейс на незалежні частини, які можна багаторазово використовувати, і подумати про кожен фрагмент окремо.

Атрибути (Props) дозволяють передавати користувацькі дані до компонента і працюють як спосіб, яким використовується для встановлення значення за замовчуванням при відображенні компонентів JSX.

Об'єкт стану (State) – це місце, де ви зберігаєте значення властивостей, що належать компоненту. Коли об'єкт стану змінюється, компонент перебудовується.

HTML5 – це остання еволюція стандарту, який визначає HTML. Термін представляє дві різні концепції. Це нова версія мови HTML з новими елементами, атрибутами та поведінкою, а також більшим набором технологій, що дозволяє створювати більш різноманітні та потужні веб-сайти та додатки.

Цей набір іноді називають HTML5 і друзі та часто скорочують до HTML5. За допомогою HTML5 браузері, такі як Firefox, Chrome, Explorer, Safari та інші, можуть знати, як відображати певну веб-сторінку, знати, де знаходяться елементи, куди розміщувати зображення та де розміщувати текст.

Основною перевагою HTML5 перед попередником (HTML4) є те, що це дає можливість додавати мультимедійний вміст без використання Flash або іншого медіаплеєра. Завдяки HTML5 користувачі можуть отримувати доступ до веб-сайтів, не підключаючись до Інтернету. До цього додається функція перетягування, а також онлайн-редагування документів, яке популяризували Google Docs.

Використовуючи теги `<video>` та `<audio>` HTML5 дає можливість додавати мультимедійні елементи без використання Adobe Flash або будь-якого іншого стороннього плагіна.

HTML-код можна легко розділити між тегами та вмістом, що дозволяє розробнику працювати ефективніше та швидше виявляти помилки.

У HTML5 є кілька нових функцій та нових елементів. Деякі з найважливіших – це summary, time, aside, audio, command, data, datalist, тощо.

Поєднавши всі вище згаданих технологій ми маємо розробити такий графічний інтерфейс, який буде реалізовувати діаграму варіантів використання наведену в Додатку Ж.

Для створення звіту був розроблений відповідний React компонент.

```
import React from 'react';
import styles from '../Report.module.scss';
import { IReport, IReportTrendEvent, IReportTrendRunSegment } from
'types/Report.type';
import VerticalInfoCard from 'partials/VerticalInfoCard/VerticalInfoCard';
import BackBtn from 'partials/BackBtn/BackBtn';
import TabHeader from 'partials/TabHeader/TabHeader';

interface IProps {
  report: IReport;
  onCardClick: (reportTrend: IReportTrendEvent | IReportTrendRunSegment) => void;
}
const InsuarenceReport: React.FC<IProps> = ({ report, onCardClick }) => {
  return (
    <>
      <BackBtn />
      <TabHeader header={reportHeader(report)} subHeader={report.description}
styleHeader={{ whiteSpace: 'pre' }} />
      <div className={styles.grid}>
        {report.items?.length &&
          report.items.map((reportTrend, index) => (
            <VerticalInfoCard
              key={reportTrend.uuid}
              reportTrend={reportTrend}
              rate={index + 1}
              callback={onCardClick}
            />
          ))}
      </div>
    </>
  );
};
export default InsuarenceReport;
```

За рахунок повторного використання компонентів стає можливим уникнути дублювання коду та пришвидшити подальшу розробку.

Використавши бібліотеку Redux [19] для керування станом додатку ми створимо компонент для профіля користувача.

```
import { call, put, takeEvery } from 'redux-saga/effects';
import { toast } from 'react-toastify';

import { ApiUrl } from '../constants/apiEndpoints';
import { IProfile } from '../types/Profile.type';
import { http } from '../util/apiServiceRequest';
import { createActions, createReducers, createActionTypes } from
'../util/reduxBoilerplate';
import { createLoadingState } from '../util/valueState';
import { getStore } from 'store/configureStore';

const resource = 'PROFILE';
const actions = ['get', 'patch', 'uploadPicture', 'changePassword', 'reset'];
type reducerType = IProfile;

export const profileTypes = createActionTypes(resource, actions);
export const profileActions = createActions<reducerType>(resource, actions);
export const profileReducer = createReducers<reducerType>(resource, actions);

export interface IProfileUpdatePayload {
  firstName: string;
  lastName: string;
}

interface IProfileUpdateAction {
  payload: IProfileUpdatePayload;
}

function* getProfile() {
  try {
    yield put(profileActions.get.commit(createLoadingState()));

    const response = yield call(http.get, ApiUrl.profile);
    yield put(profileActions.get.commit(response));
  } catch (err) {
    yield put(profileActions.get.fail(err));
  }
}
```

Таким чином будуть створені усі наступні компоненти та будуть використані повторно в тих місцях системи де це буде потрібно. Такий підхід дасть змогу значно зменшити кількість коду та запобігти його дублюванню.

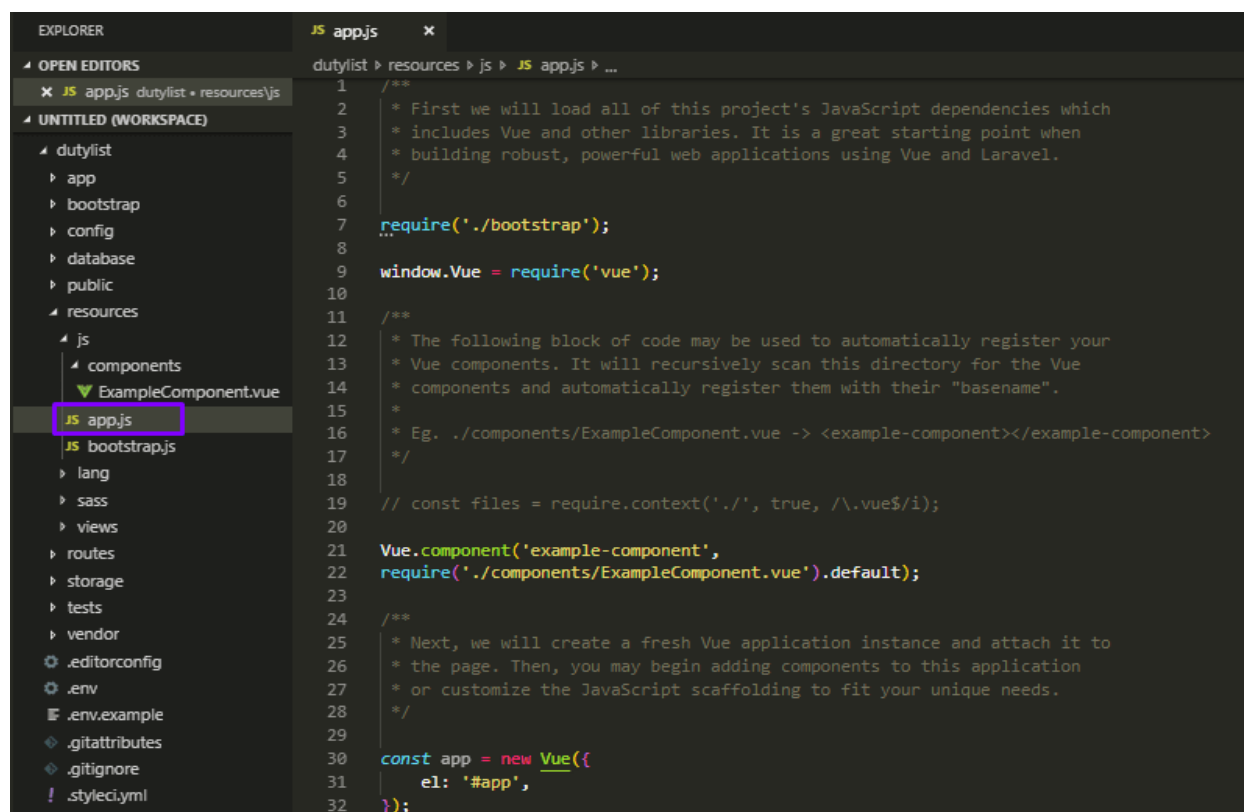
### 3.7 Розробка Backend

Laravel – це сучасний та структурований фреймворк для веб-додатків, який регулярно очолює списки найкращих фреймворків, доступних на сьогодні.

Частково це пов'язано з тим, що він базується на PHP, яка сьогодні управляє 80% Інтернету, а точка входу в мову є відносно невеликою (незважаючи на те, що вона забезпечена складними функціями, ви можете легко зрозуміти основні поняття).

За замовчуванням програми Laravel містять компонент Vue `ExampleComponent.vue`, розташований у каталозі `resources / js / components`. Файл `ExampleComponent.vue` [15] – це приклад одного файлового компонента Vue, який визначає свій шаблон JavaScript та HTML у тому самому файлі.

Однофайлові компоненти забезпечують дуже зручний підхід до побудови програм, керованих JavaScript. Приклад компонента приклад компонента (рис. 3.38)



### Рисунок 3.37 – Додаток за замовчуванням

Класи моделей Laravel повинні розширювати `Illuminate \ Database \ Eloquent \ Model`. Розташування за замовчуванням для моделей – каталог `/ app`.

Клас моделі можна легко створити командою `Artisan`. `Artisan` - це назва інтерфейсу командного рядка, що входить до складу Laravel. Він надає ряд корисних команд для використання під час розробки програми. Він керується потужним компонентом консолі `Symfony`.

Отже нам знадобиться команда: `php artisan make:model [ModelName]`. Це створить новий файл PHP у програмі [16] / за замовчуванням, який називається `[ModelName].php`, і міститиме всі шаблони для нашої нової моделі, що включає клас, простір імен та залежності, необхідні для базового налаштування.

Для створення файлу міграції разом із нашою Моделлю, використаємо таку команду: `php artisan make: model [ModelName] -m`

Де ключ `-m` також сформує файл міграції.

На додаток до створення моделі, це створює міграцію бази даних, яка підключена до моделі.

PHP-файл міграції бази даних розміщений за замовчуванням у базі даних `/ migrations /`. За замовчуванням він не включає нічого, крім стовпців `id` та `created_at / updated_at`, тому нам потрібно буде відредагувати файл, щоб надати додаткові стовпці.

Далі потрібно запустити міграцію (після налаштування файлу міграції), щоб модель почала працювати за допомогою `php artisan migrate` з кореня проекту.

Крім того, є можливість додати міграцію пізніше, після створення моделі, для цього потрібно виконати наступну команду:

```
php artisan make: migration [назва міграції]
```

Моделі можна зберігати де завгодно завдяки `PSR4`. За замовчуванням моделі створюються в каталозі програми з простором імен `App`. Для більш складних додатків зазвичай рекомендується зберігати моделі у окремих папках, яка має сенс для архітектури нашої програми.

Дотримуючись підходу «configuration negotiation». Розширюючи базовий клас Model, усі моделі успадковують властивості базові властивості, наприклад, protected \$connection, protected \$table, і т.д.

Для кожної таблиці бази даних, спроектованої раніше створимо відповідну модель. А після виконаємо міграцію для безпосереднього оновлення самих таблиць в MySQL.

Eloquent – це стандартна ORM для Laravel (об'єктно-реляційне відображення). Eloquent робить безболісним отримання і зберігання даних в нашій базі даних, використовуючи чітко визначені «моделі». Зазвичай, кожна Eloquent модель однозначно відповідає одній таблиці бази даних.

Реалізація модального вікна для коефіцієнтів виконана з використанням шаблонізатора Blade:

```
<div class="col-sm-12" id="k1-form">
  <div class="modal-header">
    <h4 class="modal-title"><i class="fa fa-plus-square"></i> Create K1</h4>
  </div>
  <div class="modal-body">
    <div class="row">
      <div class="col-sm-10 col-sm-offset-1">
        <div class="form-group">
          <label class="control-label small">K1 Name<span>*</span></label>
          <input type="text" class="form-control" name="k1_name"/>
          <label id="k1_name-error" class="error" for="k1_name"></label>
        </div>
      </div>
    </div>
  </div>
  <div class="row">
    <div class="col-sm-10 col-sm-offset-1">
      <div class="form-group">
```

```

<labelclass="control-label small">Coefficient<span>*</span></label>
<input type="number" class="form-control" name="k1_coefficient"/>
<label id="k1_coefficient-error" class="error" for="k1_coefficient"></label>
</div>
</div>
</div>
<div class="modal-footer">
    <button class="col-sm-offset-1 pull-left btn btn-primary" id="k1-create"
type="submit" data-style="zoom-in">Save</button>
    <a href="#close-modal" rel="modal:close" class="col-sm-offset-1 pull-left btn btn-
white" type="button" data-dismiss="modal">Close</a>

```

Для сутності User, в першу чергу потрібно використати модель, відповідно до таблиці users. Однак, якщо зайти в папку app проекту, можна побачити, що Laravel вже поставляється в комплекті з моделлю user, тому не потрібно створювати її вручну.

Для Eloquent [7] нам потрібно зв'язати їх. наприклад, наш User може мати кілька Params, в той час як Params прив'язаний до одного User. Визначення взаємозв'язку дозволить нам швидко проходити через наші відносини:

```

$user = App\User::find(1);
foreach ($user->params as $task) {
    echo $task->name;
}

```

По-перше, треба визначити відношення для моделі User. Відносини Eloquent визначені як методи моделей.

Eloquent підтримує декілька різних типів відносин. ми визначимо функцію equations в моделі User, яка викликає Eloquentметод hasMany ():

```

<? Php
namespace App;
// імпортування простору імен ...
class User extends Model implements AuthenticatableContract,

```

```

    AuthorizableContract,
    CanResetPasswordContract
{
    use Authenticatable, Authorizable, CanResetPassword;

    public function equations ()
    {
        return $ this-> hasMany (Task :: class);
    }
}

```

Для зв'язку user потрібно визначити відношення user для моделі params. І знову ми визначимо відношення як метод моделі. У цьому випадку ми будемо використовувати Eloquent-метод belongsTo (), що визначає зв'язок:

```

<? Php
namespace fa-fa;
use fa-fa \ User;
use fa-fa \ Params;
use fa-fa \ Coefficient
use Illuminate \ Database \ Eloquent \ Model;
class Task extends Model
{
    /**
     * Масово встановлюються атрибути.
     * /
    protected $ fillable = ['name'];
    public function user ()
    {
        return $ this-> belongsTo (User :: class);
    }
}

```

Для того щоб створити відповідні таблиці в базі даних використаємо міграції.

Міграції – це тип контролю версій для нашої бази даних. Вони дозволяють модифікувати схему бази даних та бути в курсі поточного стану схеми. Міграції, як



правило, поєднуються з конструктором схем, щоб легко керувати схемою додатка.

Щоб створити міграцію, необхідно використати команди:

migrate: make на CLIS Artisan:

```
php artisan migrate:make create_users_table
```

Міграція буде розміщена у наступній директорії `app/database/migrations` і міститиме мітку часу, яка дозволяє фреймворку визначати порядок міграцій.

Для вказання шляху до міграції треба вказати параметр `--path` під час створення.

Шлях повинен бути відносно кореневого каталогу нашої інсталяції:

```
php artisan migrate:make foo --path=app/migrations
```

Параметри `--table` та `--create` також можуть використовуватися для зазначення назви таблиці та того, чи буде міграція створювати нову таблицю:

```
php artisan migrate:make add_params_version_to_user_table --table=users
php artisan migrate:make create_users_table --create=users
```

Запуск усіх видатних міграцій:

```
php artisan migrate
```

Запуск усіх міграцій для директорії:

```
php artisan migrate --path = app/fa-fa/migrations
```

Запуск усіх міграцій пакету:

```
php artisan migrate --package=vendor/package
```

Деякі операції міграції є руйнівними, тобто вони можуть призвести до втрати даних. Щоб захистити нас від запуску цих команд проти нашої волі на виробничій базі даних, перед виконанням цих команд нам буде запропоновано підтвердження. Для того щоб змусити команди запускатись без перевірки, потрібно використати прапор `--force`:

Для реалізації входу користувачів у систему було розроблено шаблон з використанням технології `blade`:

```
@extends('layouts.auth')
@section('content')
<div class="middle-box text-center loginscreen">
  <div>
    <div>
      
```

```

</div>
<h3>Admin panel</h3>
<form class="m-t" role="form" method="POST" action="{{ route('login') }}">
    {!! csrf_field() !!}
    <input type="hidden" name="remember" value="1">

    <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}">
        <input id="email" type="email" class="form-control" name="email" value="{{ {
old('email') }}"
        placeholder="Email" required autofocus>

        @if ($errors->has('email'))
            <span class="help-block">
                <strong>{{ $errors->first('email') }}</strong>
            </span>
        @endif
    </div>

    <div class="form-group{{ $errors->has('password') ? ' has-error' : '' }}">
        <input id="password" type="password" class="form-control" name="password"
placeholder="Password" required>

        <button type="submit" class="btn btn-primary block full-width m-
b">Login</button>

</form>
<p class="m-t"> <small>© 2020 FaFa Club.</small>

```

Laravel також включає простий спосіб наповнити базу даних тестовими даними за допомогою seed класів. Усі класи seed зберігаються в додатку app/database/seeds. Класи seed можуть мати будь-яке ім'я.

## 4 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

У даному розділі виконано аналіз та розробку стартап-проекту відповідно до теми дисертації.

### 4.1 Опис ідеї проекту

В таблиці (табл. 4.1) наведено зміст ідеї, можливі напрямки до застосування продукту стартапу та основні вигоди, що може отримати користувач цього продукту.

Таблиця 4.1 – Опис ідеї стартап проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
	Бізнес що безпосередньо займається авто-страхуванням чи є посередником у наданні цих послуг.	Розширення клієнтської бази за рахунок унікального сервісу з обліку сервісних робіт автотранспорту.
	Бізнес що працює в дотичній до авто-страхування сфері. Наприклад мережа СТО.	Можливість розширити свій бізнес та отримати додаткових клієнтів та прибуток.

Використаємо порівняльну таблицю порівняння і аналізу показників: для власної ідеї визначаються показники, що мають а) гірші значення (W, слабкі); б) аналогічні (N, нейтральні) значення; в) кращі значення (S, сильні) (табл. 4.2).

Таблиця 4.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№	Техніко-економічні характеристики ідеї	Мій проект	Конкуренти			W	N	S
			1	2	3			
1	Вибір типу транспортного засобу	+	-	-	+			+

2	Вибір об'єм двигуна	+	+	+	-		+	
3	Врахування Стажу керування (в т.ч безаварійного)	+	-	+	-		+	
4	Врахування року випуску	+	-	-	+			+
5	Врахування право на пільги	+	-	-	-			+
6	Чи використовується як таксі	+	+	-	+			+
	Облік сервісних робіт	+	-	-	-			+
7	Період страхування	+	-	-	-			+

#### 4.2 Технічний аудит ідеї проекту

Проведемо аудит технологій, за допомогою яких буде реалізовано ідею. Результати цього аудиту наведені в табл. 4.3.

Таблиця 4.3 – Технологічна здійсненність ідеї проекту

№	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1		VirtualBox	Наявна	У відкритому доступі
2		Ubuntu 20.04	Наявна	У відкритому доступі
3		Apache2	Наявна	У відкритому доступі
4		ReactJS	Наявна	У відкритому доступі
5		HTML5	Наявна	У відкритому доступі
6		Laravel	Наявна	У відкритому доступі

Виходячи із даних таблиці, всі технології є у вільному доступі і стартап є технологічно здійснений.

#### 4.3 Аналіз ринкових можливостей запуску стартап-проекту

Аналіз ринку буде наступним кроком в таблиці 4.4.

Таблиця 4.4 – Попередня характеристика потенційного ринку проекту

№	Показники стану ринку	Характеристика
1	Кількість головних гравців, од	3-5
2	Загальний обсяг продаж, грн/ум.од	Середня кількість договорів укладена складає ~200 тис. ум.од
3	Динаміка ринку	Постійно зростає
4	Наявність обмежень для входу	Присутня конкуренція, звертають увагу на рішення відомих компаній
5	Специфічні вимоги до стандартизації та сертифікації	Закон України Про страхування

В табл. 4.5 наведено групи клієнтів, їх потреби та вимоги до продукту.

Таблиця 4.5 – Характеристика потенційних клієнтів стартап-проекту

№	Потреба, що формує ринок	Цільова аудиторія	Відмінності у поведінці різних потенційних цільових груп	Вимоги споживачів до товару

1	Потреба у розширенні та залученні клієнтів за рахунок нового каналу продажів та додаткових послуг.	Середній бізнес, що розглядає варіанти розширення, розвитку та впровадження нових каналів реалізації страхових продуктів.	Відсутність гнучкого інструменту для коректного та якомога точного розрахунку страхового продукту. Залежність від обмеженого функціоналу.	Швидкість. Краща ціна. Розширеній функціонал підбору. Можливість обліку сервісних робіт автотранспорт у.
---	--	---	---	--

Визначивши потенційні групи клієнтів проведено аналіз ринкового середовища на фактори загроз (табл. 4.6) та можливостей (табл. 4.7).

Таблиця 4.6 – Фактори загроз

№	Фактор	Зміст загрози	Можлива реакція компанії
1	Конкуренція	Вже існуючі компанії на ринку	Додавання нових безкоштовних сервісів у додаток. Зменшення маржі.
2	Розвиток технологій	Впровадження нових технологій, що не використовуються в системі.	Випуск нових версій системи з новими можливостями для клієнтів.

Таблиця 4.7 – Фактори можливостей

№	Фактор	Зміст можливості	Можлива реакція компанії
1	Залучення інвестицій	Можливість залучення коштів як від фондів так і від компаній партнерів	Можливість своєчасного випуску оновлень, розширення штату, та розвитку системи.
2	Обмежений функціонал. Відсутність системи обліку робіт.	Обмежений функціонал аналогів у конкурентів, відсутність можливості вести облік сервісних робіт автотранспорту.	Розширення та впровадження нового функціоналу для більш гнучкого та вчасного обліку сервісних робіт автотранспорту.

У табл. 4.8 наведено загальні риси конкуренції на ринку

Таблиця 4.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції: чиста	Значна кількість гравців на ринку	Впровадження додаткового безкоштовного функціоналу.
2. За рівнем конкурентної боротьби: національна	Гравці працюють на національному ринку	Необхідність конкурувати на національному ринку
3. За галузевою ознакою: внутрішньогалузева	Робота сконцентрована на одній галузі – страхування	Сконцентруватися на удосконаленні сервісу продажу страхових послуг

4. Конкуренція за видами товарів: товарно-видова	Значна кількість конкурентів у сфері	авторитет має набуватися шляхом бездоганного сервісу та гнучкого інтерфейсу системи
5. За характером конкурентних переваг: нецінова	Невеликі цінові відмінності між гравцями ринку	Забезпечити високий рівень сервісу та додати безкоштовні сервіси.
6. За інтенсивністю: марочна	Конкуренти виступають під своїм брендом	Запровадження авторитетного на впізнаваного бренду

Використовуючи модель М. Портера, зробимо аналіз умов конкуренції (табл. 4.9).

Серед наявних прямих конкурентів на ринку були обрані продукти Oh.UA, Finscanner, Polismart, що займають вагому частку ринку та становлять найбільшу загрозу.

Таблиця 4.9 – Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти в галузі	Постачальники	Клієнти	Товари-замінники
	Oh.UA, Finscanner, Polismart	Банки	Polismart Finscanner	Середній бізнес	Незначна кількість
Висновки	Значна конкурента боротьба	За умов конкуренто здатного сервісу можливий вихід на ринок враховуючи	Постачальники обмежені в функціоналі та додаткових сервісах	При високій конкуренції клієнти висувають додаткові вимоги	Відсутність локальних замінників з ідентичним рівнем сервісу



		високу щільність гравців			
--	--	--------------------------------	--	--	--

Виходячи із попередніх даних, видно, що в системи є можливість працювати на ринку, проте нам потрібно визначити конкурентоздатну ціну та характеристик продукту.

В табл. 4.10 наведено обґрунтування факторів конкурентоспроможності

Таблиця 4.10 – Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Гнучкий та широкий функціонал розрахунку страхових коефіцієнтів	Цей функціонал виділяю систему поміж конкурентів та дозволяє якомога точніше та економічно обґрунтовано для кінцевого клієнта розраховувати кінцеву вартість.
2	Облік сервісних робіт автотранспорту	Це унікальний сервіс системи який дозволяє утримувати клієнтів та спонукати їх до повторного використання системи.
3	Гнучкий адаптивний інтерфейс	Використовуючи інтерфейс системи який адаптований під усі сучасні девайс ми можемо розраховувати на більшу кількість клієнтів.

Далі виконаємо порівняльний аналіз сильних та слабких сторін продукту (табл. 4.11)

Таблиця 4.11 – Порівняльний аналіз сильних та слабких сторін

№	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів						
			-3	-2	-1	0	1	2	3

1	Гнучкий та широкий функціонал розрахунку страхових коефіцієнтів	15		x					
2	Облік сервісних робіт автотранспорту	18			x				
3	Гнучкий адаптивний інтерфейс	12				x			

SWOT-аналіз (табл. 4.12) є фінальним кроком

Таблиця 4.12 – SWOT-аналіз стартап-проекту

Сильні сторони: гнучкий та широкий функціонал розрахунку страхових коефіцієнтів; облік сервісних робіт автотранспорту	Слабкі сторони: новий гравець на ринку
Можливості: отримання клієнтів що прагнуть низьких та прозорих цін та клієнтів що хочуть отримувати повний сервіс автотранспорту в одному місці.	Загрози: достатньо щільна конкуренція на ринку

На основі аналізу розроблено альтернативи ринкової поведінки та перелік заходів з їх впровадження (табл. 4.13).

Таблиця 4.13 – Альтернативи ринкового впровадження стартап-проекту

№	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Зацікавлення конкурентів	Реалізація системи чи її поглинання конкурентами	Декілька років
2	Пошук інвесторів із дотичних сфер	Пропозиція інвесторам із сфер пов'язаних з автотранспортом	До року

3	Окрема реалізація системи обліку сервісних робіт автотранспорту	Окремий розвиток систем обліку сервісних робіт автотранспорту. Цей сервіс може зацікавити власників СТО.	До року
---	---	--	---------

Вибраною альтернативою є пошук інвесторів із дотичних сфер.

Розроблення ринкової стратегії проекту

В табл. 4.14 наведено вибір потенційних груп споживачів продукту. Такими групами стали малий та середній бізнес.

Таблиця 4.14 – Вибір цільових груп потенційних споживачів

№	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Малий бізнес та стартапи	Не готові	Не високий, адже ця група сама знаходиться в активній фазі розвитку	Не висока	Складний
2	Середній бізнес	Готові	Високий попит за рахунок конкретної ціни та унікальності сервіса	Висока	Високий

Обрано групи середнього бізнесу, адже малий бізнес та стартапи не має достатнього потенціалу для впровадження.

Для роботи в обраному сегменті ринку необхідно визначити базову стратегію розвитку (табл. 4.15).

За її результатом було визначено, що обраною альтернативою став підхід продажу свого рішення за меншою ціною.

Таблиця 4.15 – Визначення базової стратегії ринку

№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Продаж рішень	Низька ціна та підтримка платформ з меншою конкуренцією	Зменшення собівартості товару та концентрація на потребах невеликих груп користувачів	Концетрований маркетинг

У табл. 4.16 визначено базову стратегію конкурентної поведінки майбутнього продукту.

Таблиця 4.16 – Визначення базової стратегії конкурентної поведінки

№	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Ні	Пошук нових споживачів серед невеликих компаній та спроба забирати частку у вже існуючих	Так, але вносити унікальні властивості	Стратегія заняття конкурентної ніші

Далі визначимо стратегію позиціонування (табл. 4.17).

Таблиця 4.17 – Визначення стратегії позиціонування

№	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Гнучкий та широкий функціонал розрахунку страхових коефіцієнтів; облік сервісних робіт автотранспорту гнучкий адаптивний інтерфейс	Аналіз та провадження функціоналу відсутнього у конкурентів. Реалізація унікального сервісу з обліку сервісних робіт автотранспорту.	Обґрунтована та низька ціна на страхові продукти. Гнучкий адаптивний інтерфейс. Унікальний сервіс обліку сервісних робіт автотранспорту.	Гнучкість ціни, адаптивний інтерфейс, облік сервісних робіт автотранспорту

#### 4.4 Розроблення маркетингової програми стартап-проекту

Сформулюємо маркетингову концепцію продукту, що отримає споживач. Результат аналізу конкурентоспроможності наведено в табл. 4.18.

Як видно з наведених даних, на даний момент на ринку немає конкурента який включає в себе ті переваги що надає наша система.

Таблиця 4.18 – Визначення ключових переваг концепції потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
---	---------	----------------------------	--

1	Прозорий та економічно обґрунтований підбір страхового продукту	Реалізована можливість підбору страхового продукту з урахуванням максимально можливої кількості показників	Дає можливість економити на підборі поліса, враховує всі можливі знижки.
---	---	--	--

Розробимо трирівневу маркетингову модель товару (табл. 4.19).

Таблиця 4.19 - Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові	
1. Товар за задумом	Реалізація системи онлайн-страхування та обліку сервісних робіт автотранспорту	
	Властивості/характеристики	Розмір
	HUB Сервіс	38 Мб
	State API	20 Мб
	RDS Service	100 Мб
	Proxy service	12 Мб
	Якість: відбувалось тестування за допомогою unit та integration тестів у процесі розробки	
	Пакування: відсутнє	
	Марка: назва компанії та версія продукту	
	До продажу: код продукту	
	Після продажу: набір .html, .js, .php файлів	
За рахунок чого потенційний товар буде захищено від копіювання: ліцензування та договору про заборону копіювання		

Визначимо цінові межі для ціноутворення (табл. 4.20)

Таблиця 4.20 – Визначення меж встановлення ціни

№	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	\$6500-17000	\$9000-14000	>\$400000	\$0-\$1500

Збут буде виконуватись власними силами та наведений у таблиці 4.21

Таблиця 4.21 – Формування системи збуту

№	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Придбання ліцензії на програмний продукт	Підписання контрактів на збут	Однорівневий	Вертикальна, за якою право власності залишається у розробника продукту

Фінальним кроком стане визначення концепції маркетингових комунікацій (табл. 4.22)

Таблиця 4.22 – Концепція маркетингових комунікацій

№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Одноразова купівля за	Рекламні інтеграції.	Реалізація систем онлайн-	Представлення унікального	Зазначення сильних

	допомогою підписання контракту	Розповсюдження через клієнтів.	страхування та обліку сервісних робіт автотранспорту	робочого функціонала системи	сторін продукту серед конкурентів
--	--------------------------------	--------------------------------	--	------------------------------	-----------------------------------

#### 4.5 Висновки

У даному розділі був проведений аналіз та створення порівняльних таблиць стартап-проекту, було описано слабкі та сильні сторони проекту, встановлено технічну можливість його реалізації.

Також було визначено, що на ринку присутня висока конкуренція з боку інших компаній, одна їх сервіси обмежені у функціоналі та не мають системи обліку сервісних робіт автотранспорту.

Була напрацьована стратегія ринку з орієнтацією на середній ринок, адже він виглядає найбільш привабливий з точки зору впровадження продукту.



## ВИСНОВКИ

Пересічному громадянину важко зорієнтуватися у великій кількості пропозицій та полісів, що пропонують страхові компанії на сьогоднішній день. Тому постає питання необхідності розробки програмного забезпечення, яке б допомогло потенційному клієнту обрати саме те, що йому потрібно не докладаючи для цього зайвих зусиль.

Метою магістерської дисертації є автоматизація та вдосконалення процесу формування і реалізації страхових полісів, що дасть змогу розраховувати конкурентну вартість страхових продуктів шляхом врахування всіх можливих коефіцієнтів та зменшення штату працівників.

При виконанні аналізу предметної області та існуючих рішень з'ясувалося, що кількість сервісів для купівлі страхових полісів онлайн щороку збільшується, проте далеко не всі вони здатні повністю задовольнити потреби клієнта. Так, в результаті огляду найпопулярніших сучасних рішень онлайн страхування (Oh.UA, Finscanner.com, Polismart.com.ua) було вирішено розробити власну систему для підбору та купівлі страхових полісів онлайн.

За основу/зразок для розробки власної системи прийнято сервіси Oh.UA, Finscanner.com, Polismart.com.ua. Вирішено, що веб-додаток повинен складатися з двох модулів: модуля підбору страхової компанії і купівлі страхового полісу та модуля обліку сервісних робіт і витрат транспортного засобу.

Запропонована система онлайн страхування та обліку сервісних робіт автотранспорту «Fa-Fa Club» має низку переваг порівняно з іншими сервісами.

По-перше, модуль системи з підбору та реалізації страхового поліса має наступні переваги:

- вибір надійної страхової компанії за рейтингом;
- порівняння страхових компаній;
- купівля страхового полісу онлайн за найбільш вигідною ціною;
- вибір способу доставки полісу у будь-яку точку України.

По-друге, в системі реалізований модуль обліку сервісних робіт автотранспорту, який буде надзвичайно привабливим для власників транспортних засобів, що підлягають обов'язковому технічному контролю. Цей модуль дає можливість:

- налаштування графіку обов'язкового технічного контролю;
- нагадування про необхідність проведення сервісних робіт;
- слідкування за станом авто;
- оновлення пробігу для виконання регламентних робіт на його основі;
- облік витрат на транспортний засіб (витрати на паливо, ремонт тощо);
- список точок-партнерів, де можна здійснювати на пільгових умовах ремонт, шиномонтаж та інші види сервісних робіт, з адресами, маршрутами і можливістю бронювання;
- максимально швидке та зручне бронювання, замовлення та онлайн оплата послуг у вищезазначених точках.

Результати магістерської дисертації були використані у впровадженні системи реалізації страхових продуктів та сервісних робіт, що підтверджуються актом впровадження.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Управление корпоративной ИТ-инфраструктурой / А.И. Ролик, С.Ф. Теленик, М.В. Ясочка // К.: Наукова думка, 2018. – 576 с.
2. Скамай Л.Г. Страхование. Теория и практика. Учебник. - Москва: ЮРАЙТ, 2014 - 384 с.
3. - <http://zakon5.rada.gov.ua/laws/show/1961-15> - Назва з екрану
4. Бретт Кинг. Банк online 3.0. Почему сегодня банк – это не то, куда вы ходите, а то, что вы делаете. - Москва: ООО "Олимп-Бизнес", 2016. - 520 с.
5. Закон України "Про обов'язкове страхування цивільно-правової відповідальності власників наземних транспортних засобів" [Електронний ресурс]: Режим доступу Golden Krishna. The Best Interface is No Interface. - New Riders, 2015. - 39 с.
6. jQuery [Електронний ресурс]: Режим доступу - <https://jquery.com/> - Назва з екрану.
7. Laravel Framework [Електронний ресурс]: Режим доступу - <https://laravel.com/docs/5.4> - Назва з екрану.
8. Web-сервер Apache2 [Електронний ресурс]: Режим доступу <https://httpd.apache.org/> - Назва з екрану.
9. Джон Резиг, Расс Фергюссон, Джон Пакстон. JavaScript для профессионалов, 2-е изд. - Москва: ООО "И.Д. Вильямс", 2016. - 240 с.
10. Дейт К. Введение в системы баз данных. - К.; М.; СПб.: Изд. дом "Вильямс", 2000. - с.
11. Николай Прохоренок, Владимир Дронов. HTML, JavaScript, PHP и MySQL. Джентльменский набор
12. Web-мастера, 4-е издание. - Санкт-Петербург: БХВ-Петербург, 2015. - 766с.
13. SOAP Message Exchange Patterns [Електронний ресурс] – Режим доступу до ресурса: <https://www.w3.org/TR/soap12-part1/#soapmer>

14. AWS. Break the Monolith [Электронный ресурс] – Режим доступа до ресурсу: <https://aws.amazon.com/ru/getting-started/container-microservices-tutorial>
15. From Design to Deployment / Chris Richardson, Floyd Smith, 2016. – 74
16. Sandeep Kumar Patel. Developing Responsive Web Applications with AJAX and jQuery. - Packt Publishing, 2014. - 248 pages.
17. Мелансон Б., Нордин Д., Луиси Ж. Профессиональная разработка сайтов на Drupal 7. - Санкт-Петербург: Питер, 2013. - 288 с.
18. Денис Колесниченко. PHP и MySQL. Разработка Web-приложений, 5-е издание. - Санкт-Петербург: БХВ-Петербург, 2015. - 593с.
19. Грофф Дж. Р., Вайнберг П.Н., Оппель Э. Дж. SQL. Полное руководство. 3-е издание. - Москва: ООО "И.Д. Вильямс", 2015. - 959 с.
20. ECMAScript 2019 Language Specification [Электронный ресурс] – Режим доступа до ресурса: <https://www.ecma-international.org/ecma-262/10.0/index.html>
21. Node.js – a JavaScript runtime [Электронный ресурс] – Режим доступа до ресурса: <https://nodejs.org/>
22. Bit. Easily share components across projects and applications as a team. [Электронный ресурс] – Режим доступа до ресурсу: <https://bit.dev/>
23. The Top JavaScript Frameworks For Front-End Development in 2020 [Электронный ресурс] – Режим доступа до ресурсу: <https://www.freecodecamp.org/news/complete-guide-for-front-end-developers-javascript-frameworks-2019/>